A STUDY OF METAL CUTTING BY VISCOPLASTIC FINITE ELEMENT FORMULATION AND COMPARISON WITH EXPERIMENTS

A Thesis Submitted
In Partial Fulfilment of the Requirements
for the Degree of
MASTER OF TECHNOLOGY

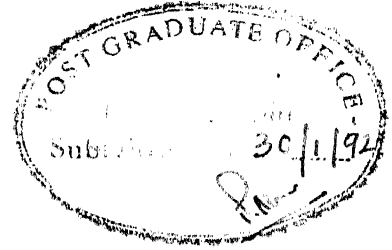
by
ANAND S. DESHPANDE

to the
DEPARTMENT OF MECHANICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY KANPUR
JANUARY, 1992

DEDICATED

TO

MY BELOVED FATHER



CERTIFICATE

It is certified that the work contained in the thesis entitled "A STUDY OF METAL CUTTING BY VISCOPLASTIC FINITE ELEMENT FORMULATION AND COMPARISON WITH EXPERIMENTS" by Anand Sharad Deshpande has been carried out under our supervision and that this work has not been submitted elsewhere for a degree.

(Dr. G.K. Lal)
Professor
Dept. of Mechanical
Engineering
I.I.T. Kanpur

(Dr. T. Sundararajan)
Asst. Professor
Dept. of Mechanical
Engineering
I.I.T. Kanpur

Feb. 1992

TH
671.3
D459s

16 MAR 1992
CENTRAL LIBRARY
KANPUR
Acc. No. A113088

ME-1992-M-DES-STU

ACKNOWLEDGEMENT

I express my deep sense of gratitude to Dr. T. Sundararajan for suggesting and supervising the present work. I am extremely thankful to him for his inspiring guidance, meticulous attention, constructive criticism and above all for his untiring devotion throughout the tenure of this work. I must mention that I "learnt" finite element technique and theoretical modelling through his simple and interesting explainations.

I am extremely thankful to Dr. G.K. Lal for the constant encouragement he gave to me during this work. I am indebted to him especially for giving me valuable suggestions while analysing the metal cutting process understanding viscoplasticity concepts and experimental tests conducted.

Thanks to Mr. Jha, Mr. Sharma of Manufacturing Science lab for helping me during experimentation.

I am very thankful to my friends Venkat Reddy, Arun Yadav, Sudipto Ghosh, Ranen Bhattacharya and many others for helping me on the various occasions during the present work.

I must not forget to say thanks to my wife Rekha, my mother and Milind.

Thanks to all those who helped me during my stay in IIT Kanpur.

IIT Kanpur
February 1992

A.S. Deshpande

ACKNOWLEDGEMENT

I express my deep sense of gratitude to Dr. T. Sundararajan for suggesting and supervising the present work. I am extremely thankful to him for his inspiring guidance, meticulous attention, constructive criticism and above all for his untiring devotion throughout the tenure of this work. I must mention that I "learnt" finite element technique and theoretical modelling through his simple and interesting explainations.

I am extremely thankful to Dr. G.K. Lal for the constant encouragement he gave to me during this work. I am indebted to him especially for giving me valuable suggestions while analysing the metal cutting process understanding viscoplasticity concepts and experimental tests conducted.

Thanks to Mr. Jha, Mr. Sharma of Manufacturing Science lab for helping me during experimentation.

I am very thankful to my friends Venkat Reddy, Arun Yadav, Sudipto Ghosh, Ranen Bhattacharya and many others for helping me on the various occasions during the present work.

I must not forget to say thanks to my wife Rekha, my mother and Milind.

Thanks to all those who helped me during my stay in IIT Kanpur.

IIT Kanpur
February 1992

A.S. Deshpande

C O N T E N T S

ABSTRACT

In the present work, the thermal and deformation phenomena during metal cutting have been analyzed, using the visco-plastic deformation model. The Finite element technique has been employed to solve the coupled stress balance, mass balance and energy balance equations. Since the governing equations for the visco plastic deformation of a material are similar to those of a non-Newtonian, incompressible fluid, the flow approach has been used for solution purposes. The velocity, pressure and temperature fields thus predicted are further processed to obtain the isothermal patterns, deformation patterns and the cutting forces which are of practical importance. Also, a parametric study has been conducted to investigate, the effects of various process parameters such as depth of cut, cutting velocity and rake angle on the temperature and deformation patterns in the plastic zone. One of the significant contributions of this analysis is that important factors such as thermal softening of material, variation of material properties with temperature etc. have been considered. A complete software with pre-processing, FEM analysis and past-processing has been developed as a part of the study. A number of experiments for force and temperature measurements in orthogonal metal cutting have been conducted. Using the results of these experiments as well as some results of the earlier researchers, the theoretical model has been validated. Also, the present theoretical model confirms the energy minimization principle proposed by Merchant for estimating the chip thickness or shear angle.

NOMENCLATURE

$\underset{\sim}{d}$, σ'_{ij}	Deviatoric stresses
$\underset{\sim}{\sigma}$	Stress tensor
$\underset{\sim}{\dot{e}}$	Strain Rate tensor
$\underset{\sim}{f}$	Body Force vector (N/m^2)
p	Mean Normal Stress (N/m^2)
g	Acceleration due to gravity (m/S^2)
t	Depth of cut (mm)
t_c	Chip thickness (mm)
$\underset{\sim}{V}$	Velocity vector (m/s)
V	Cutting velocity (m/s)
V_c	Chip velocity (m/s)
u	Velocity in x direction (m/s)
v	Velocity in y direction (m/s)
x	coordinate in cutting direction.
y	coordinate Normal to the cutting direction
ξ	Local x-coordinate
η	Local y-coordinate
α	Rake angle (degrees)
ϕ	Shear angle (degrees)
μ	Dynamic Viscocity ($N-s/m^2$)
ρ	Density (kg/m^2)
σ_Y	Uniaxial yield stress (N/m^2)
σ_{yo}	Uniaxial yield stress at atmospheric temperature (N/m^2)
σ_{ij}	Stress component
e_{ij}	Strain component
$\bar{\tau}$	Effective yield shear stress (N/m^2)
τ_F	Effective frictional shear stress (N/m^2)
m	Friction Factor

k	Thermal conductivity of workpiece material (W/mK)
k_o	Thermal conductivity of workpiece at atmospheric temprature (w/mK)
k_t	Thermal conductivity of tool material (W/mK)
C_p	Soecific Heat of workpiece material
C_{po}	Specific heat of Workpiece material at atmorpheric temperature (KJ/KgK)
α	Thermal Diffusivity (m^2/sec)
h	Overall heat transfer coefficient (W/m^2K)
P_e	Peclet Number
T	Absolute temperature
T_{amb}	Ambient temperature.
$\dot{Q}$	Volumetric heat generation rate (W/m^3)
$\dot{Q}_F$	Frictional heat generation rate (w/m^3)

OPERATORS

L	A differential operator
$\underset{\sim}{\nabla}$	Gradient operator
∇^2	Double differential operator
∂	Partial differential operator.

Super Script

$*$	Non dimensionalized quantity

LIST OF FIGURES AND TABLES

CHAPTER I

INTRODUCTION

1.1 GENERAL BACKGROUND

Metal cutting forms the basis of a wide variety of machining processes employed in modern engineering industry. Nearly half of the engineering products today are produced through machining at one stage or the other. In this age of wide spread automation, we are witnessing the trend that the demand for productivity and process optimization is increasing day by day. In view of the crucial role played by machining among all the material-processing operations, a scientific understanding of the metal cutting process is vital for modern engineering technology and practice.

A detailed analysis of the mechanics of metal-cutting which sheds light on the underlying plastic deformation processes and thermal phenomena, is helpful in the prediction of some of the input parameters such as cutting forces. Evaluation of cutting forces is of paramount importance for the following reasons :

(i) to estimate the power requirement of a machine tool

(ii) to select appropriate bearings, jigs and fixtures which can withstand the tool forces

(iii) to survey the characteristics of new work and tool materials by investigating tool forces.

Further, minimization of the power input for a certain metal removal rate (MRR) often forms the objective of optimizing the metal cutting operation. Also, an insight into the effects of various process variables like the tool geometry, tool material

properties, cutting velocity and temperature distribution near tool-tip region is useful in understanding processes such as the formation of built-up edge, tool wear and tool vibration. Minimizing such undesirable effects can increase productivity and lower machines costs.

Over a past few decades, a large volume of experimental data has been collected on various aspects of metal cutting, but a comprehensive theoretical understanding of this highly complex process is yet to emerge. In order to estimate some of the important parameters such as cutting force, one has to rely on empirical relations based on experimental measurements and material properties.

The present work, in its right earnest, is a small effort to fill some existing gaps in the theory of metal cutting. A coupled viscoplastic analysis of the deformation and thermal processes in the vicinity of the tool-tip has been attempted here, to provide a reasonably accurate estimate of the most important input parameters such as cutting forces and the process variables such as chip thickness ratio and tool-tip temperature. The theoretical predictions have been compared against experimental results obtained as a part of the present study and also with those of earlier researchers, to validate the theoretical model.

1.2 REVIEW OF PREVIOUS WORK

The earliest systematic study of metal cutting process, dates back to the end of 19th century, when pioneering researchers like Mallock, Tresca etc attempted to throw some light on the mechanics of machining. However, a major contribution to this

area which has an important influence on the practical development of machining was possible only when Taylor [1] published his work in 1907. He was particularly concerned with tool wear and tool life and seems to have been one of the first to recognize the influence of temperature on tool wear. He developed a relationship between tool life and speed of machining which is still in use today.

After this work, the major contribution in the area of metal cutting commenced from the post war period. The first attempts towards developing an understanding of the mechanics of metal cutting were performed by analysing the chip-formation process. As early as 1945, a pioneering work by Merchant [2] and Piispanen [3] was presented on the so called classical "single shear plane model". This model assumes that the chip is formed due to plastic deformation occurring along a single plane, known as the shear plane. The most advantageous aspect of this model is that it enables the determination of the average yield shear stress and slip velocity along the shear plane, purely through simple geometric constructions. It also established beyond any doubt that metal cutting is basically a shear deformation process. The Merchant-Piispanen model was subsequently improved by Palmer & Oxley [4] and Okushima & Hitomi [5]. These studies suggested that though the formation of chip occurs due to flow of the metal under shear, the deformation is not limited to a single plane; it occurs in a narrow region approximated by two parallel planes. This model later came to be known as the thick shear zone model. In view of the small thickness of the shear zone in comparison to its length, this model assumes that the stake of stress within the

deformation region is uniform simple shear.

Around the same period of the above mentioned theoretical models, efforts were going on to validate these models with experimentation. Keceloglu [6] was the first person to observe the process through photo-micrographs in 1958. The plastic zone where the chip formation takes place was photographed at short intervals using a quick stop device. By noting the deformation of the grain boundaries, the size and the shape of plastic deformation zone were estimated. This study established that the plastic region could be approximately represented by a thin parallel-sided zone. Later in 1969, a more effective way of using photo-micrograph technique was proposed by Stevenson and Oxley [7]. Assuming a thin parallel-sided zone, they observed printed grids on the material before and after deformation. From the streamlines of metal flow, the strains and strainrates were calculated. The important process variable of flow stress (or effective yield stress) could also be evaluated as a function of strain by this method, for a wide range of strainrates. Later investigators [8,9] proposed that the flow stress is a strong function of strainrate as well.

J.T. Black [8] proposed an approach based on the theory of dislocation mechanics. He suggested that the flow stress is composed of two parts i.e. thermal and non-thermal. The observations of the study indicated that the plastic zone is divided into alternate shear and lamella bands. The shear bands (or shear fronts) give rise to the dominant non-thermal part while the lamella regions lead to a thermal part of the flow stress. Evaluation of flow stress was based on the stacking fault energy

(SFE). This approach, mostly used by metallurgists, is not very useful to the continuum mechanics based theories.

Von Turkovich [9, 10] first observed peculiar form of plastic flow, extremely localised in a small region in the chip ahead of the tool edge along the rake face. This was called the secondary shear deformation zone. The experimental studies conducted so far, have established that material behaves like a highly viscous fluid within the deformation zones and as a perfect solid (with infinite viscocity and zero strain rate) outside them. The viscocity of such a fluid, depends strongly on material deformation rate.

The credit for the effective usage of the FEM technique to viscoplastic analysis goes to Zienkiewicz et.al. [11, 12, 13]. In 1973, these researchers applied FEM for the modeling of metal forming and extrusion problems. In metal cutting, the FEM technique was employed by Tay and Stevenson [14, 15] for predicting the temperature field for orthogonal cutting. They used hyperbolic streamlines in the primary deformation zone for calculating velocity fields and strainrates. They also derived semi empirical relations for computing heat generation. However, these studies do not provide a coupled analysis of plastic deformation and thermal processes during metal cutting.

Muraka and Hinduja [16] predicted the temperature field variation during metal cutting and verified the average temperature rise by experimental measurements. These authors obtained the temperature distribution near the tool tip for a wide range of cutting conditions. Balaji [17] applied the FEM

technique to evaluate the temperature field within the work material and the tool, for coated carbide tools. He also, has provided the experimental verifications of the results obtained. Mallikarjun Sarma [18] used the FEM-based viscoplastic analysis t/o predict velocity, pressure and temperature fields. But, his work did not consider some of the major aspects such as thermal softening and the variation in the material properties such as thermal conductivity, specific heat, yield stress etc. Still, it gave a reasonably good estimate for the temperature field and the size of the primary deformation zone. In the most recent times, Komvopowrs etal. [31] have analyzed the deformation processes in metal cutting using elasto-plastic approach. The deformation pattern obtained here seem to reasonably good but, the thermal aspects as well as the experimental validation and estimation of important quantities such as forces is not attempted here also.

Most of the FEM solutions available at present on metal cutting do not provide a complete theoretical analysis developed from the first principles. It is also observed that hardly any FEM work speaks of the parameters of practical importance such as cutting forces. Also very often gross simplifications have been invoked which are necessitated by the complexities of the process as well as the domain. These simplifications obviously affect the accuracy of the solution. In the present work, some of the limitations of the existing FEM solutions have been eliminated.

1.3 OBJECTIVES AND SCOPE OF PRESENT STUDY

In the present study, an attempt has been made to fill the gap in the analysis of metal-cutting by solving the coupled

flow and energy equations. Stresses in the cutting zone have been modeled by considering visco-plastic material behaviour during deformation. Accordingly, the metal flow beyond the elastic limit is treated similar to that of a liquid of very high viscocity. Viscocity itself is described as a strong, non-linear function of the strain-rate and yield stress of the work material. The material is considered to be purely strain-rate sensitive, with no effects of work-hardening. It must also be noted that material properties such as specific heat, thermal conductivity and uniaxial yield stress vary with change of temperature. Metal cutting is bound to involve large temperature variations. Hence, to have a realistic picture of the process, the variations in material properties with respect to temperature have been considered. In particular, the effect of thermal softening has been incorporated by considering the reduction in the value of yield stress with increase in temperature.

It is well established from previous experiments that the deformation is restricted to a very small region around the tool tip. Therefore, a solution domain extending up to a few millimeters from tool-tip into the workpiece and the chip has been considered. Beyond this domain, the material is assumed to be perfectly rigid. The tool is also taken to be perfectly rigid and sharp with no built up edge. The tool cutting edge is assumed to be orthogonal to the cutting direction. Further, for generating the domain and finite element mesh easily, the geometry of the problem has been simplified in the following manner. The curved top surface of the chip is given by a hyperbola with asymptotic parallel to the feed direction and the rake face. The bottom

surface on the other hand, is in perfect contact with the rake face upto the contact point and it becomes parallel to the top surface beyond the point of contact. The chip thickness is determined as a part of the analysis by invoking the minimum total energy principle. The inlet and exit boundaries are taken to be orthogonal to the feed direction or the rake face. Though the chip curling process makes the real boundaries of the problem very complex, the simplifications employed in the present study are appropriate, as the focus of this investigation is to perform coupled stress and heat transfer analysis for a given domain.

Another important assumption invoked in the present study is that all the powers expended in plastic deformation is converted into heat. Though some residual stresses/strains are retained both in the workpiece and the chip, this assumption simplifies computation to a great extent without sacrificing the accuracy of predictions, when compared with experimental results.

The governing equations of flow and energy after the incorporation of the above mentioned assumptions, have been solved using the Finite Element Technique. The velocities, temperature and pressure distributions are obtained and utilized to identify the shapes of isotherms and for the prediction of cutting forces. Also from the strainrates obtained in the domain, a reasonably good idea of the dimensions of primary and secondary deformation zone is obtained.

A special software for FEM grid generation has been developed in the present study, since grid generation for this complex geometry encountered in the metal cutting problem is quite

a difficult task. Some of the advanced concepts such as the hyperbolic streamline model proposed by Tay and Stevenson [15] trans-finite interpolation etc. have been used in the grid generation algorithm. To have a good overview of the predicted results, the analysis has been carried out for various cutting conditions viz. different depths of cut, cutting velocities and rake angles. A number of experiments on orthogonal cutting have been performed to verify the results obtained from the theoretical analysis.

After developing a complete package consisting of the finite element analysis and grid generation, the effect of variation in chip thickness on the total energy consumed during metal cutting has been studied. This can be viewed as an excellent check for the present theoretical approach, since Merchants [1] theory uses the energy minimization principle while evaluating the chip thickness or shear angle.

CHAPTER 2

THEORETICAL FORMULATION

Metal cutting involves very complex deformation phenomena. Extremely localised asymmetric deformation occurs at exceedingly high strains and strainrates, in a minute domain of few millimeters size. The metal cutting action itself is a consequence of high compression followed by shearing of the metal. Consequently, any theoretical treatment of the metal cutting process is bound to possess certain relaxations and simplifications.

A two dimensional orthogonal machining under dry cutting conditions has been considered for analysis. In fact, orthogonal cutting is a particular case of oblique cutting, with cutting edge of the tool perpendicular to the cutting velocity vector. The tool is assumed to be perfectly sharp, with no builtup edge. The work material is taken to be ductile, resulting in the formation of continuous chips. The tool and the portion of work material outside the solution domain on the other hand, are taken to be perfectly rigid without any deformation.

2.1 Plastic deformation and chip formation in orthogonal machining

In orthogonal metal cutting, the fact that the width of the chip is large compared to the its thickness, renders the problem two-dimensional. Although the deformation in metal cutting is similar to those of the metal forming processes a distinguishing feature is that the deformed material, is separated from the parent workpiece in the form of a chip. Hence, it is of

prime importance to analyze the chip formation process, when the theoretical analysis of metal cutting is done.

The uncut material in motion is first interrupted by the cutting tool, causing severe compressive stresses in the material. After sufficient compression, when the resulting shear stresses in the material attain the yield value, plastic flow of the material occurs in the direction of shear forces; leading to the formation of the chip. The outward or shearing movement of each successive deforming element is arrested by workhardening and the movement is transferred to the next element.

It is well accepted now that the predominant mechanism of metal cutting is the shearing process occurring in the zone called "Shear Zone". A detailed study of metal cutting reveals that two such zones exist in the domain.

i) Primary shear deformation zone (PSDZ)

ii) Secondary shear deformation zone (SSDZ)

These two zones have been shown in Fig. (2.1). The primary shear zone has been observed to be wedge shaped and its thickness mainly depends upon cutting velocity. It has been reported that an increase in cutting velocity causes the thickness of PSDZ to reduce. The thickness or mean width of PSDZ is an important quantity which gives an idea of the rate of deformation and the amount of heat generated. The narrower the region, the greater the rate of deformation.

The secondary shear deformation zone lies adjacent to the contact patch between the chip and the tool. Additional deformation due to the friction takes place here. Each layer in SSDZ moves at a different speed because of shearing in a direction

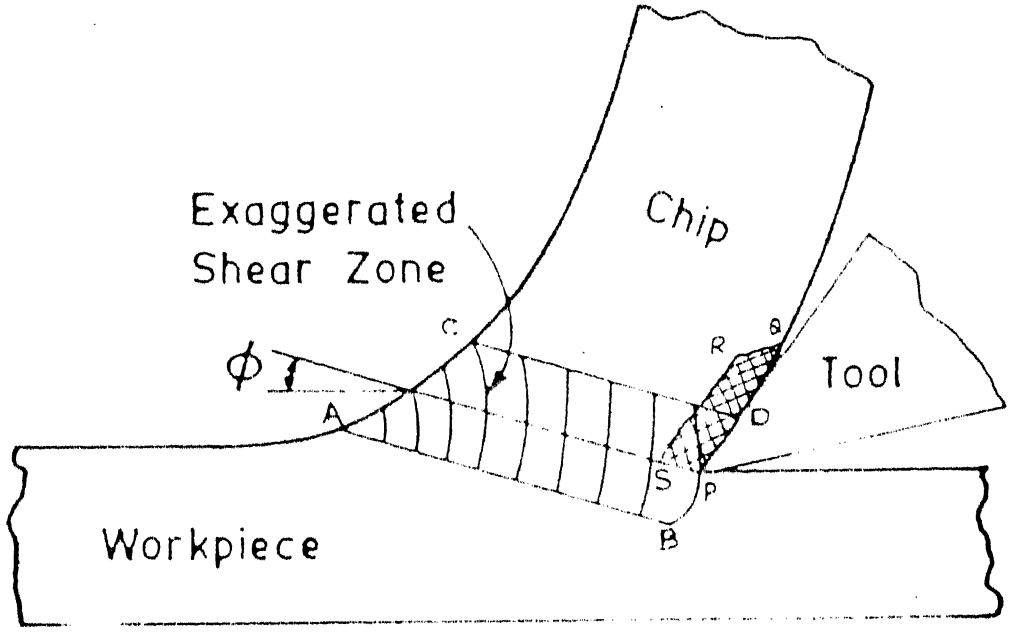


Fig. 2.1 Shear Zones During Metal Cutting.

parallel to the rake face. Although the region is small as compared with PSDZ, the amount of heat produced is believed to be of a similar magnitude. An interesting point to note is that, the highest temperature occurs on the interface between SSDZ and rake face. The temperature in SSDZ is higher than in PSDZ mainly because, the metal flow velocity is very low in SSDZ and also the material which enters SSDZ has already been heated through PSDZ. Some of the researchers have attempted to estimate the thicknesses of these zones theoretically and experimentally [27].

2.2 Viscoplasticity Concepts

The study of the behaviour of a ductile material under tensile load reveals that beyond the elastic limit when the material yields plastically, the stress (some times called as flow stress) becomes a function of strain-rate. Some of the properties of the solid undergoing plastic deformation are akin to that of a Non-Newtonian Fluid in the sense that, the relationship between the shear stress and the shear strainrate is non-linear.

The material incompressibility is often assumed in the analysis of plastic deformation processes and is quite reasonable since, yield occurs primarily due to shear. Therefore, the sum of all the diagonal components of strain-tensor (which represents volumetric strain) is zero.

i.e. $$e_{ii} = e_{11} + e_{22} + e_{33} = 0 \qquad (2.1)$$

Deviatoric stress and deviatoric strain or strainrate are of vital importance in theory of plasticity. These are defined as

$$e'_{ij} = e_{ij} - 1/3\ e_{ii}\delta_{ij} \qquad (2.2)$$

$$\sigma'_{ij} = \sigma_{ij} - 1/3\ \sigma_{ii}\delta_{ij} \qquad (2.3)$$

where

e_{ii} = volumetric strain

e_{ij} = strain tensor

e'_{ij} = deviatoric strain tensor

σ_{ij} = stress tensor

$\sigma_{ii/3}$ = isotropic stress

σ'_{ij} = deviatoric stress.

Various theories of plasticity developed so far, attempt to relate the deviatoric part of strain (or strainrate) with deviatoric stress.

An extensively used yield criterion in the analysis of plastic deformation processes is due to Von Mises. According to this criterion, the second invariant of deviatoric stress tensor (represented by J_2) determines the criterion of yielding. Mathematically, yield occurs when

$$J_2 = 1/2\ \sigma'_{ij}\ \sigma'_{ij} \geq K^2 \qquad (2.4)$$

where, K is a material yield constant.

In the plastic deformation, the total strain deviation consists of two parts i.e. elastic and plastic. But, in the problems such as metal cutting or metal forming involving huge plastic deformations, the elastic strain deviation makes a contribution of just 2 to 3% only and can therefore be neglected.

Hence, $$e'_{ij} \approx e'^{(p)}_{ij} \qquad (2.5)$$

The total plastic strain $e^{(p)}_{ij}$ occurring for a loading period of time t is given by the area under the curve of strain-rate ($\dot{e}_{ij}$) and time (t). Thus,

$$e^{(p)}_{ij} = e^{(p)}_{ij}\ (0) + \int_0^t \dot{e}_{ij}\ dt \qquad (2.6)$$

where $e^{(p)}_{ij}(0)$ is the plastic strain at the initial time t = 0. From the principle of material incompressibility (from eqs. 2.1-2.5)

$$e^{(p)}_{ii} = 0 \qquad (2.7)$$

The strainrate tensor $e^{(p)}_{ij}$ is thus, same as $e'^{(p)}_{ij}$. Sine, only the rate of plastic strain is specified by above equations for a varying loading sequence, the total plastic strain $e^{(p)}_{ij}$ can often be replaced by the algebraic sum of successive strain increments.

The addition of all the minute strain increments to obtain the total strain forms the essence of the the incremental theory of plasticity.

2.3 Mathematical Formulation

2.3.1 Constitutive relation for viscoplastic flow

The metal undergoing plastic deformation can be described as a Von-Mises type of strain-rate sensitive, non-strain hardening, visco-plastic material. This, in turn, is similar to

an incompressible, non-newtonian fluid for which the above mentioned viscoplasticity concepts can be easily applied.

An important aspect to be noted here is that, the viscocity of such a "Fluid" is dependent on the current local strainrates, total accumulated plastic strain undergone by the material and the local temperature. It has been assumed in the present analysis that work hardening is absent. Thus, for a purely strainrate sensitive material, the constitutive relation is

$$\sigma'_{ij} = 2\mu \dot{e}'_{ij} \tag{2.8}$$

where, μ = viscocity.

$\dot{e}'_{ij}$ = deviatoric part of strain rate tensor

The strainrate tensor, for a two dimensional case can be written as

$$\dot{e}_{11} = \partial u/\partial x \tag{2.9a}$$

$$\dot{e}_{22} = \partial v/\partial y \tag{2.9b}$$

$$\dot{e}_{12} = \dot{e}_{21} = 1/2 \left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right] \tag{2.9c}$$

where, u and v are the velocity components in x and y directions.

The viscocity μ for a strainrate sensitive material can be written in the form of the constitutive relation given by

$$\mu = \frac{\sigma_y + \left[\dfrac{\bar{\dot{e}}}{\sqrt{3}\,\gamma}\right]^{1/n}}{\sqrt{3}\;\bar{\dot{e}}} \tag{2.10}$$

where σ_y = yield stress of the material

γ & n = Physical constants which define the viscoplastic characteristics of the material.

Effective yield stress which is often called as Flow stress ($\bar{\sigma}$) is a function of the uniaxial yield stress (σ_y) and strain-rate ($\dot{\bar{e}}$). Thus,

$$\bar{\sigma} = \sigma_y + \left[\frac{\dot{\bar{e}}}{\sqrt{3}\,\gamma} \right]^{1/n} \quad . \tag{2.11}$$

From eqn. (2.10) and (2.11) we get,

$$\bar{\sigma} = \sqrt{3}\,\mu\dot{\bar{e}} \quad . \tag{2.12}$$

The strain rate invariant $\dot{\bar{e}}$ can be expanded in terms of strain rate tensor as,

$$\dot{\bar{e}} = \sqrt{2\,\dot{e}_{ij}\,\dot{e}_{ij}} \quad . \tag{2.12}$$

For a two dimensional case, above eqn., (2.12) reduces to,

$$\dot{\bar{e}} = \sqrt{2(\dot{e}_{11}^2 + \dot{e}_{12}^2 + 2\dot{e}_{12}\dot{e}_{21})} \tag{2.13}$$

i.e. $$\dot{\bar{e}} = \sqrt{2\left\{\left(\frac{\partial u}{\partial x}\right)^2 + \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2\right\}} \tag{2.14}$$

2.3.2 Governing Equations :

In the present analysis, the metal cutting problem is being viewed with the perspective of the viscoplastic deformation process. Especially, the velocity fields, pressure field and the temperature field in the plastic domain need to be predicted.

From the consideration of material balance, stress balance and heat balance, the steady state governing equations for the velocity, pressure and temperature fields are given by :

$$\nabla . \underset{\sim}{V} = 0 \quad \text{.....(From mass balance)} \tag{2.15}$$

$$\rho(\underset{\sim}{V} . \nabla \underset{\sim}{V}) = \nabla . \underset{\approx}{\sigma} + \underset{\sim}{f} \quad \text{(From stress balance)} \tag{2.16}$$

and

$$\nabla .(k \nabla T) + \dot{Q} = \rho C_p \ (\underset{\sim}{V} . \nabla T) \quad \text{(From energy balance)} \tag{2.17}$$

where ρ = Density of the workpiece material

$\underset{\sim}{V}$ = The velocity vector

$\underset{\approx}{\sigma}$ = Stress tensor

$\underset{\sim}{f}$ = Body Force tensor

k = Thermal conductivity of work-material

T = Temperature

$\dot{Q}$ = Heat generation rate per unit volume.

C_p = Specific heat of workpiece material.

Flow Equations :

The stress tensor in the momentum equation can be expressed as,

$$\underset{\approx}{\sigma} = - p \underset{\approx}{I} + \underset{\approx}{d} \tag{2.17}$$

where, p is the pressure and $\underset{\sim}{d}$ is the deviatoric part of the stress tensor.

The deviatoric stress tensor $\underset{\approx}{d}$ can be written as,

$$\underset{\approx}{d} = \mu(\nabla \underset{\sim}{V} + \nabla \underset{\sim}{V}^T) - \frac{2}{3} (\nabla . \underset{\sim}{V}) \underset{\approx}{I} \tag{2.18}$$

where, $\nabla \underset{\sim}{V}^T$ is the transpose of $\nabla \underset{\sim}{V}$.

Simiplifying eqn. (2.18), using incompressibility condition, we

can write,

$$\underset{\approx}{\sigma} = -p\underset{\approx}{I} + \mu\,(\nabla \underset{\sim}{V} + \nabla \underset{\sim}{V}^{T}) \qquad (2.19)$$

Thus, the momentum equation gets the form,

$$\rho\,(\underset{\sim}{V}.\nabla \underset{\sim}{V}) = \nabla.\left\{-p\underset{\approx}{I} + \mu\,(\nabla \underset{\sim}{V} + \nabla \underset{\sim}{V}^{T})\right\} \qquad (2.20)$$

where, the body force vector $\underset{\sim}{f}$ (which usually is the weight of the material per unit volume) has been absorbed in to the pressure.

Component Form of Flow equations

Material balance equation for a two dimensional case can be written as,

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \qquad (2.21)$$

where, u and v are the velocity components in x and y directions.

The stress tensor $\underset{\approx}{\sigma}$ in a two dimensional case, can be expressed in component from as :

$$\underset{\sim}{\sigma} = \sigma_{xx}\,\hat{i}\hat{i} + \sigma_{xy}\,\hat{i}\hat{j} + \sigma_{yx}\,\hat{j}\hat{i} + \sigma_{yy}\,\hat{j}\hat{j}\,. \qquad (2.22)$$

Thus, the component forms of the momentum equation can be written as follows :

X-momentum

$$\rho\left[u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right] = \left\{\frac{\partial}{\partial x}(\sigma_{xx}) + \frac{\partial}{\partial y}(\sigma_{yx})\right\} \qquad (2.23)$$

and

Y-momentum

$$\rho\left[u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right] = \left\{\frac{\partial}{\partial x}(\sigma_{xy}) + \frac{\partial}{\partial y}(\sigma_{yy})\right\}\,. \qquad (2.24)$$

Stress tensor components are given by :

$$\left.\begin{aligned} \sigma_{xx} &= -p + 2\mu \frac{\partial u}{\partial x} \\ \sigma_{xy} &= \sigma_{yx} = \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) \mu \end{aligned}\right\} \qquad (2.25)$$

and $$\sigma_{yy} = -p + 2\mu \frac{\partial v}{\partial y} \quad .$$

Using (2.25), the final forms of momentum equations are

$$\rho\left(u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right) = \left\{ \frac{\partial}{\partial x}(-p+2\mu \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right]\right\} \qquad (2.27)$$

$$\rho\left(u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y}\right) = \left\{ \frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right)\right] + \frac{\partial}{\partial y}\left[\left(-p+2\mu \frac{\partial v}{\partial y}\right)\right]\right\} \qquad (2.28)$$

Energy Equation

The objective of solving the energy equation is to predict the temperature field. It is thus obvious that, the heat generation must be evaluated carefully. In metal cutting, it can be reasonably assumed that the heat generation is due to (i) plastic work in PSDZ, (ii) Friction in SSDZ. (also some plastic work) It has been assumed that all of the work done is converted into heat, which seems to be quite reasonable because, the material density does not change considerably and also, the work hardening is not severe due to high temperature conditions.

Volumetric heat generation rate can mathematically be expressed as,

$$\dot{Q} = \bar{\tau} \; \bar{\dot{e}} \qquad (2.29)$$

where, $\bar{\tau}$ = Effective yield shear stress.

and $\dot{\bar{e}}$ = strain rate invariant.

Using Von-Misses criterion, (2.29) can be simplified as,

$$\dot{Q} = \frac{\bar{\sigma}}{\sqrt{3}} \dot{\bar{e}} \qquad (2.30)$$

Thus, using (2.30) and (2.17), the final form of energy equation becomes,

$$k \left[\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial Y^2}\right] + \frac{\bar{\sigma}}{\sqrt{3}} \dot{\bar{e}} = \rho C_p \left[u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y}\right] \qquad (2.31)$$

2.3.3 Non-Dimensionalization :

As long as the number of parameters involved in a problem are few in number, it is realistic to stay with the dimensional form of variables, which enables ready comparison of predicted results with real values. However, when the number of variables is large, analysing the influence of each parameter on the final result becomes a tedious task. The subject of metal cutting is known to involve parameters, which exhibit very complex interdependences among themselves. To circumvent such hardships and facilitate a detailed parametric study, it is desirable to non-dimensionalize the variables of the problem with suitable scaling factors.

A real difficulty which often arises during non-dimensionalization, is the proper choice of a scaling factor for particular quantity. Though some general guidelines do exist, it can be stated that, the primary guide in this process is intuition alone, since there is no unique way for non-dimensionalization. However, the best possible way is to

give, the physics of the problem, a critical examination and incorporate as much of the physical characteristics of the problem as possible while selecting the scaling factors.

For the present problem, to begin with, the velocity components u and v are non-dimensionalized with the cutting velocity (V); the distances with depth of cut (t); and the pressure with the uniaxial yield stress at ambient temperature (σ_{yo}). The scaling factors for the remaining variables will be selected conveniently, during the course of non-dimensionalization itself.

Thus, we define

$$x^* = x/t; \qquad y^* = y/t; \quad u^* = u/V; \quad v^* = v/V$$

where * denotes a dimensionless quantity. Using the above definition in material balance equation,

$$\Sigma \frac{\partial(u^* V)}{\partial(x^* t)} + \Sigma \frac{\partial(v^* V)}{\partial(y^* t)} = 0$$

i.e.
$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} = 0 \qquad (2.32)$$

Momentum Equations

x-Momentum equation

x-momentum equation in dimensional form is given by

$$\rho\left[u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y}\right] = \left\{ \frac{\partial}{\partial x}(-p+2\mu \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}\left[\mu\left[\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right]\right]\right\} \qquad (2.33)$$

where, ρ = Density of the materil

μ = viscocity

At, this stage, the viscocity can be scaled as shown below :

$$\mu^* = \frac{\mu}{(\sigma_{yo} t/V)} = \left[\frac{\sigma_y + \left[\frac{\dot{\bar{e}}}{\sqrt{3}\,\gamma} \right]^{1/n}}{\sqrt{3}\ \dot{\bar{e}}} \right] \times \frac{1}{(\sigma_{yo} t/V)}$$

$$= \frac{(\sigma_y/\sigma_{yo}) + \frac{1}{\sigma_{yo}} \left\{ \left(\frac{V}{t}\right) \frac{1}{\sqrt{3}\gamma} \right\}^{1/n} (\dot{\bar{e}}^*)^{1/n}}{\sqrt{3}\ \dot{\bar{e}}^*}$$

or, $$\mu^* = \frac{\sigma_y^* + B^* (\dot{\bar{e}}^*)^{1/n}}{\sqrt{3}\ \dot{\bar{e}}^*} \qquad (2.34)$$

where,

$$\sigma_y^* = \sigma_y/\sigma_{yo}$$

$$B^* = \frac{1}{\sigma_{yo}} \left\{ \left(\frac{V}{t}\right) \frac{1}{\sqrt{3}\ \gamma} \right\}^{1/n} \rightarrow \text{constant for the prescribed material and cutting conditions}$$

and $\dot{\bar{e}}^* = \dot{\bar{e}}/(V/t)$

Now, non-dimensionalizing each quantity of the equation (2.33) we get

$$\rho V^2 \left[u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial y^*} \right] = \sigma_{yo} \left\{ \frac{\partial}{\partial x^*} (-p^* + 2\mu^* \frac{\partial u^*}{\partial x^*}) + \frac{\partial}{\partial y^*} \left[\mu^* \left(\frac{\partial u^*}{\partial y^*} + \frac{\partial v^*}{\partial x^*} \right) \right] \right\} \qquad (2.35)$$

or,

$$\frac{1}{\sigma}\left(u^{*}\frac{\partial u^{*}}{\partial x^{*}} + v^{*}\frac{\partial u^{*}}{\partial y^{*}}\right) = \frac{\partial}{\partial x^{*}}\left(-p^{*}+2\mu^{*}\frac{\partial u^{*}}{\partial x^{*}}\right) + \frac{\partial}{\partial y^{*}}\left[\mu^{*}\left(\frac{\partial u^{*}}{\partial y^{*}} + \frac{\partial v^{*}}{\partial x^{*}}\right)\right] \tag{2.36}$$

where, $\sigma = \sigma_{yo} / (\rho V^{2})$.

Y-Momentum equation

Similarly, on the above lines the y-momentum equation can be reduced as,

$$\frac{1}{\sigma}\left(u^{*}\frac{\partial v^{*}}{\partial x^{*}} + v^{*}\frac{\partial v^{*}}{\partial y^{*}}\right) = \frac{\partial}{\partial x^{*}}\left[\mu^{*}\left(u^{*}\frac{\partial u^{*}}{\partial y^{*}} + v^{*}\frac{\partial v^{*}}{\partial x^{*}}\right)\right] + \frac{\partial}{\partial y^{*}}\left(-p^{*}+2\mu^{*}\frac{\partial v^{*}}{\partial y^{*}}\right) \tag{2.37}$$

Energy equation

The dimensional form of energy equation in the present study is,

$$\frac{\partial}{\partial x}\left(k\frac{\partial T}{\partial x}\right) + \frac{\partial}{\partial y}\left(k\frac{\partial T}{\partial y}\right) + \dot{Q} = \rho C_{p}\left(u\frac{\partial T}{\partial x} + v\frac{\partial T}{\partial y}\right) \tag{2.38}$$

where,

k = Thermal conductivity of the work-material

C_{p} = Specific heat of the work material

$\dot{Q}$ = Heat generated.

The normalization of temperature T in this equation can be done using some reference temperature T_{ref}, as

$$T^* = \frac{T}{T_{ref}}$$

T_{ref} can be conveniently chosen as shown below. Non-dimensionalizing each quantity of euqation (2.38), we get

$$\frac{k_o}{t^2} T_{ref} \left\{ \frac{\partial}{\partial x^*}(k^* \frac{\partial T^*}{\partial x^*}) + \frac{\partial}{\partial y^*}(k^* \frac{\partial T^*}{\partial y^*}) + \frac{V}{t} \sigma_{yo} \frac{\sigma^*}{\sqrt{3}} \dot{\bar{e}}^* \right\}$$

$$= \frac{\rho V C_{po} T_{ref}}{t} \left[C_p^* (u^* \frac{\partial T^*}{\partial x^*} + v^* \frac{\partial T^*}{\partial y^*}) \right] \qquad (2.39)$$

Multiplying above equation by $t^2/(\rho C_{po})$,

$$\frac{k_o}{\rho C_{po}} T_{ref} \left\{ \frac{\partial}{\partial x^*}(k^* \frac{\partial T^*}{\partial x^*}) + \frac{\partial}{\partial y^*}(k^* \frac{\partial T^*}{\partial y^*}) \right\} + Vt \frac{\sigma_{yo}}{\rho C_{po}} \frac{\sigma^*}{\sqrt{3}} \dot{\bar{e}}^*$$

$$= Vt\, T_{ref} \left[C_p^* (u^* \frac{\partial T^*}{\partial x^*} + v^* \frac{\partial T^*}{\partial y^*}) \right] \qquad (2.40),$$

Taking, $\sigma_{yo}/(\rho\, C_{po}) = Tref$

and $k_o/(\rho\, C_{po}) = \alpha$ (Thermal diffusivity),

the final form of dimensionless heat balance euqation becpmes

$$\left\{ \frac{\partial}{\partial x^*}(k^* \frac{\partial T^*}{\partial x^*}) + \frac{\partial}{\partial y^*}(k^* \frac{\partial T^*}{\partial y^*}) \right\} + P_e \frac{\sigma^*}{\sqrt{3}} \dot{\bar{e}}^* - P_e \left[C_p^* (u^* \frac{\partial T^*}{\partial x^*} + v^* \frac{\partial T^*}{\partial y^*}) \right] = 0$$

(2.41)

where,

k^* = Non-dimensional thermal conductivity = k/k_o

C_p^* = Non-dimensional specific heat = C_p/C_{po}

P_e = Peclet Number = Vt/α

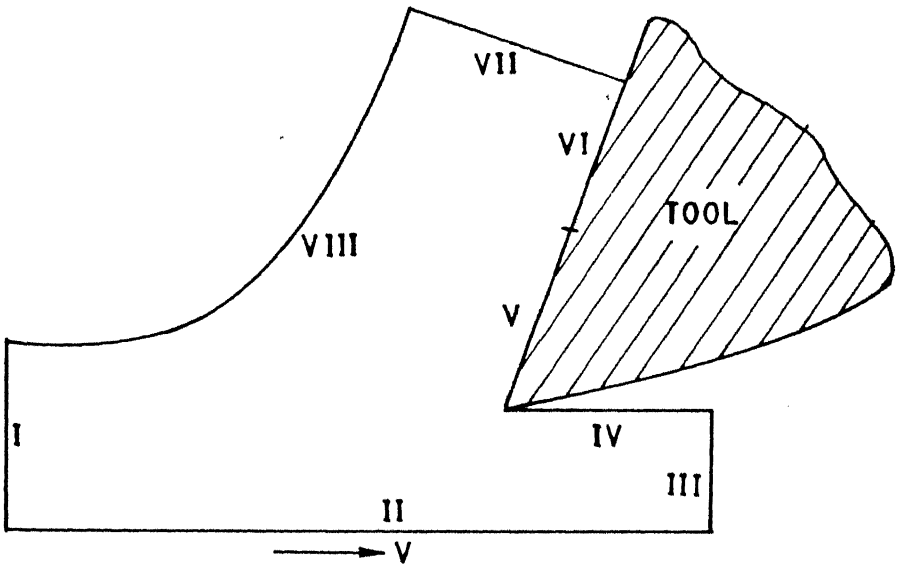


Fig. 2·2 Boundaries of the selected solution domain

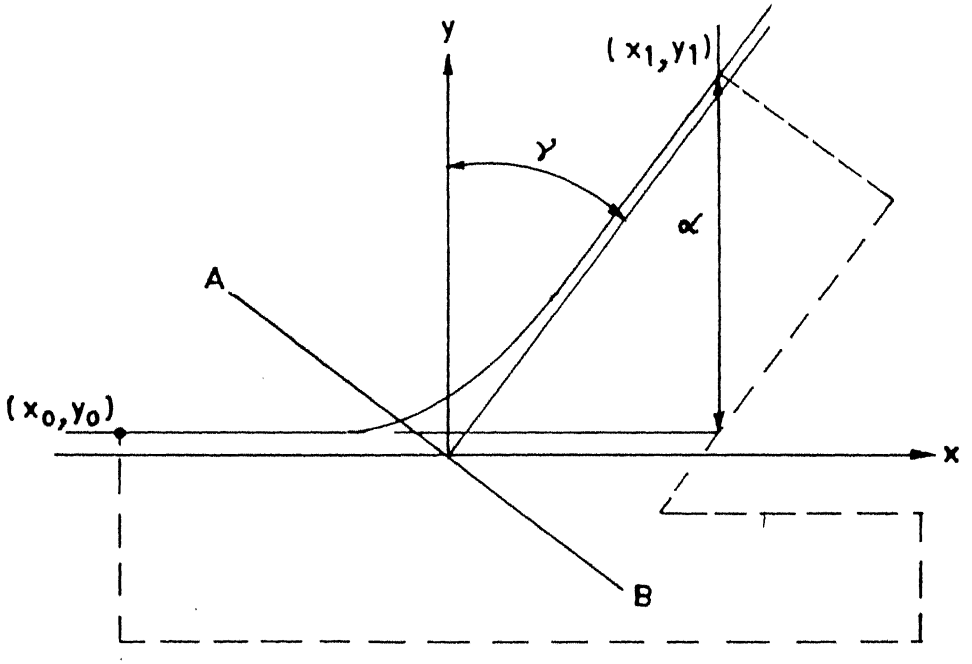


Fig. 2·3 Concept of hyperbolic streamline used in generating VIII surface of the solution domain

VII.

The free surface of the chip, where the complex chip curling effect with serrations is observed, how been modelled using the Hyperbolic streamline concept used by Tay et al. (15). This forms the surface VIII.

Hyperbolic stream line approach

This was used by Tay, Stevenson et al. (15) to find the distribution of shear strain rates in the deformation zones. The eighth surface of the present domain, very closely approximates a hyperbolic streamline used in the above work.

The hyperbolic streamline is of the form

$$Y^2 \tan \gamma - xy = a \qquad (2.6.1)$$

which is illustrated in Fig. (2.3). With this equation, the origin of the Cartesian Co-ordinate system which lies on AB is located at the intersection of the two asymptotes one of which is parallel to the cutting direction while other is inclined at an angle $(90^o - \gamma)$ to the cutting direction.

Thus, the equation for the hyperbola defined in Fig. (2.3) is given by the equation,

$$(y-mx-\bar{y})\ (y-1.0) = (y1) - mx1 - \bar{y})\ (y1 - 1.0) \qquad (2.6.2)$$

and slope

$$\frac{dy}{dx} = m(y-1.0)/[(y-1.0) + (y-mx-\bar{y})]$$

k_o & C_{po} are the thermal conductivity and specific heat of the work material at ambient temperature respectively.

2.4 Selection of Solution Domain for Viscoplastic Analysis

It is well established from previous experiments that the deformation is restricted to a very small region around the tool tip. Therefore, a solution domain extending up to a few millimeters from the tool tip into the workpiece and the chip has been considered. Beyond this domain, the material is assumed to be perfectly rigid. The tool is also taken to be perfectly rigid and sharp with no builtup edge. The tool edge is assumed to be orthogonal to the cutting direction. Further, for generating the domain and the finite element mesh easily, the boundary of the domain has been divided in to eight surfaces, as shown in Fig. (2.2). Surface I is the entry section where material enters the plastic solution domain and is considered to be plane, perpendicular to cutting velocity. Surface II, is selected in a plane parallel to the cutting velocity and restricts the boundary of the plastic zone below depth of cut. Material leaves the plastic zone at surface III which is parallel to the surface I. Surface IV is formed by the machined surface and is exposed to the atmosphere due to the the clearance angle of the tool.

The rake surface, along which perfect sticking contact exists between the chip and the tool form the V^{th} surface (sticking contact region) and rest of the contact region i.e. slipping contact region forms the surface VI.

The chip is considered to be bounded by plane parallel surfaces beyond contact point; which is represented by Surface

where, $m = \dfrac{\sin(90^{\circ}-\gamma)}{\cos(90^{\circ}-\gamma)}$

$$\bar{y} = [\bar{y}_o - \alpha y1 - (x_o - \alpha x1)m]/(1-\alpha) \qquad (2.6.3)$$

and α, x_1, y_1, x_o, y_o etc. have been defined in the Fig. (2.3)

* **Estimation of Contact Length :**

In continuous chip formation process, the chip remains in contact with the rake face of the tool over a certain length. This contact length is known as natural chip-tool contact length. The importance of the chip tool contact length in the mechanics of continuous chip formation was not realized for a long time, because of the fact that most of the theories of metal cutting, such as those developed by Merchant, Lee and Shaffer etc. did not include this as a parameter. But it has since been established that, the force of friction acting on the chip-tool interface depends upon the length of chip-tool contact.

Some amount of research is available on the contact length. Venkatesh et al. (20) have measured the contact length successfully using the Wheatstone bridge method. Almost all the theoretical analyses for the determination of contact length in natural contact cutting are based on the equilibrium of chip under the action of moments of forces acting on the tool-rate face and on the shear plane. Hahn (29) has developed a relation for the determination of the natural chip-tool contact length, making use of merchants theory as well. The values predicted by this relation : have been found to match with experimental values

reasonably, especially for the small depth of cuts of the order of 10^{-1} to 10^{-2} millimeters. The relation is given by,

$$l_c = \frac{t \sin(\phi+\beta-\alpha)}{\sin\phi \cos\beta}$$

where, l_c = Natural chip-tool contact length

t = Depth of cut

ϕ = Shear angle

β = Friction angle

α = Rake angle

In the present analysis, chip-tool contact length is an input parameter.

A simple method such as applying lamp black on the rake surface and then measuring the scar produced by the chip on it under travelling microscope, may be used for measurement of contact length. Some such measurements have been made in the present study also.

In the solution domain, surface V extends over the sticking contact length and surface VI over the slipping part of contact length. In the present work, both sticking and slipping contact length are assumed to be equal.

2.5 Boundary Conditions

The non-dimensional form of the boundary conditions are discussed below.

Refering to Fig. 2.2, for convective heat loss on boundaries IV and VIII, the dimensional for of boundary condition is

$$-k \frac{\partial T}{\partial n} = h \ (T-T_{amb}) \qquad (2.42)$$

where h = Heat transfer coefficient.

Non dimensionalizing each quantity of eqn (2.42),

$$-k^* \frac{\partial T^*}{\partial n^*} = h^* \ (T^*-T^*_{amb}) \qquad (2.43)$$

where

$$T^* = T/T_{ref}$$

$$h^* = h\ t/k_o$$

$$T^*_{amb} = T_{amb}/T_{ref}.$$

On the chip tool interface (surface V and VI), the frictional heat generated ($\dot{Q}_F$) is assumed to be partitioned according to the thermal conductivity values of the chip and the tool. The leads to the boundary conditions of the form

$$-k \frac{\partial T}{\partial n} = \left(\frac{k}{k+k_t}\right)\dot{Q}_F = \phi_k \ \dot{Q}_F \qquad (2.44)$$

where $\dot{Q}_F$ = Frictional power dissipation

$= \tau_F \times V_T$

k, k_t = thermal conductivities of the chip and the tool materials

τ_F = Effective frictional shear stress ($= \frac{m\bar{\sigma}}{\sqrt{3}}$ or $m\mu\dot{\bar{e}}$)

V_T = Tangential velocity of slip

ϕ_k = Frictional heat partition ratio

Non-dimensionalizing each quantity of (eqn. (2.44)

$$-k_o k^* \frac{T_{ref}}{t} \frac{\partial T^*}{\partial n^*} = \phi_k \times \sigma_{yo} \times \tau_F^* \times V \times V_T^* .$$

Simplification of above equation gives,

$$-k^* \frac{\partial T^*}{\partial n^*} = \phi_k \, P_e \, \tau_F^* \, V_T^* \qquad (2.45)$$

where P_e = peclet number $\left[= Vt/(k_o/\rho C_{po})\right]$

. On surfaces I, II and III

$$T^* = T_{amb}^* \qquad (2.46)$$

where, $T_{amb}^* = T_{amb}/T_{ref}$.

. At the far end of the chip (surface VII),

$$\frac{\partial T^*}{\partial n^*} = 0 \qquad (2.47)$$

This condition implies that, the temperature variation in the chip flow direction is negligible and to the absence of heat generation or any major heat loss mechanism.

. The dimnsionless velocity conditions are ;

$u^* = 1$, $v^* = 0$ on srufaces I, II, and III

$v_n^* = 0$ for surfaces IV,V, VI and VIII

The chip velocity V_c also can be non-dimensionalized as

$$V_c^* = \frac{V_c}{V}$$

resulting in

$$\left.\begin{aligned} u^* &= V_c^* \cos\alpha \\ v^* &= V_c^* \sin\alpha \end{aligned}\right\} \quad \text{For surface VII} \qquad (2.48)$$

Frictional stress condition on Surface V becomes.

$$\tau_F^* = \frac{\tau_F}{\sigma_{yo}} = \frac{m\bar{\sigma}/\sqrt{3}}{\sigma_{yo}} = \frac{m}{\sqrt{3}} \left[\frac{\sigma_y + \left[\frac{\dot{\bar{e}}}{\sqrt{3}\,\gamma} \right]^{1/n}}{\sqrt{3}\ \dot{\bar{e}}} \right]$$

$$= \frac{m}{\sqrt{3}} \left\{ \frac{\sigma_y}{\sigma_{yo}} + \frac{1}{\sigma_{yo}} \left[\frac{V}{t} \ \frac{1}{\sqrt{3}\gamma} \right]^{1/n} (\dot{\bar{e}}^*)^{1/n} \right\}$$

$$= \frac{m}{\sqrt{3}} \left\{ \sigma_y^* + B^* (\dot{\bar{e}}^*)^{1/n} \right\} \qquad (2.49)$$

$$\text{where } B^* = \frac{1}{\sigma_{yo}} \left[\frac{V}{t} \ \frac{1}{\sqrt{3}\gamma} \right]^{1/n} \qquad (2.50)$$

$\sigma_y^* = \sigma_y / \sigma_{yo}$.

m = Friction factor.

The dimensionless governing equation and boundary conditions formulated above have been solved by the application of the Finite Element Method.

2.6 Energy Minimization Principle

Chip thickness is a significant parameter of the solution domain. With the change in chip thickness or chip thickness ratio, the size of the solution domain changes.

According to Merchant (1), whose theory is based upon the 'minimum energy principle', the shear plane is located where the least energy is required for shear. This principle of

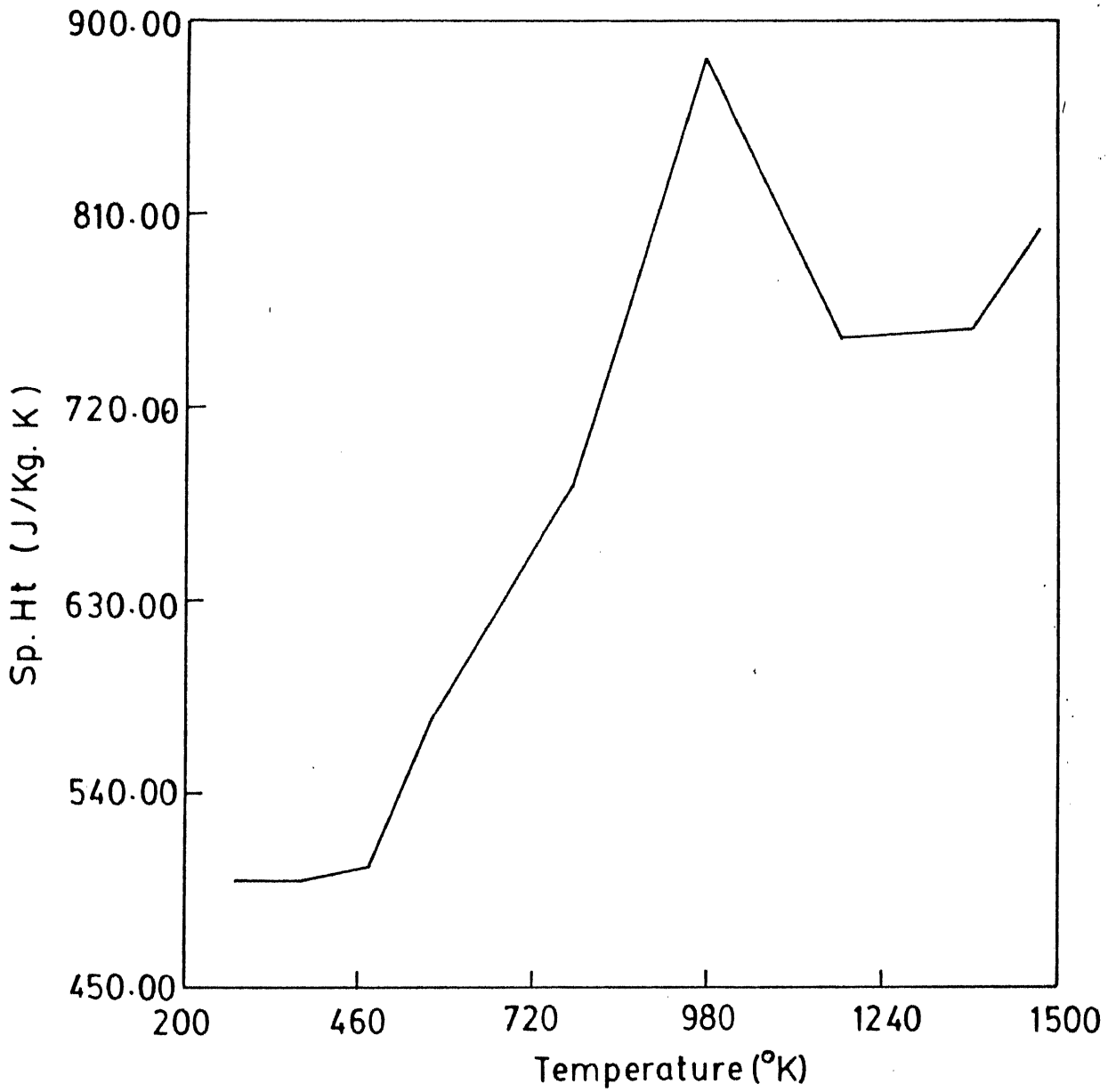


Fig. 2.4 Variation of specific heat with temperature (for MS)

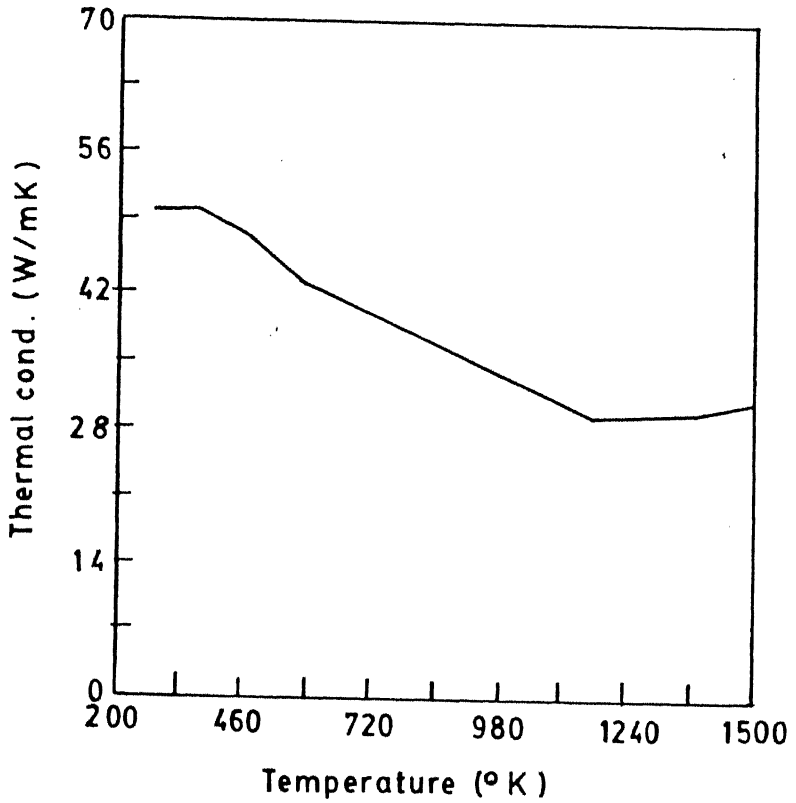


Fig. 2.5 Varriation of thermal conductivity with temperature (for MS)

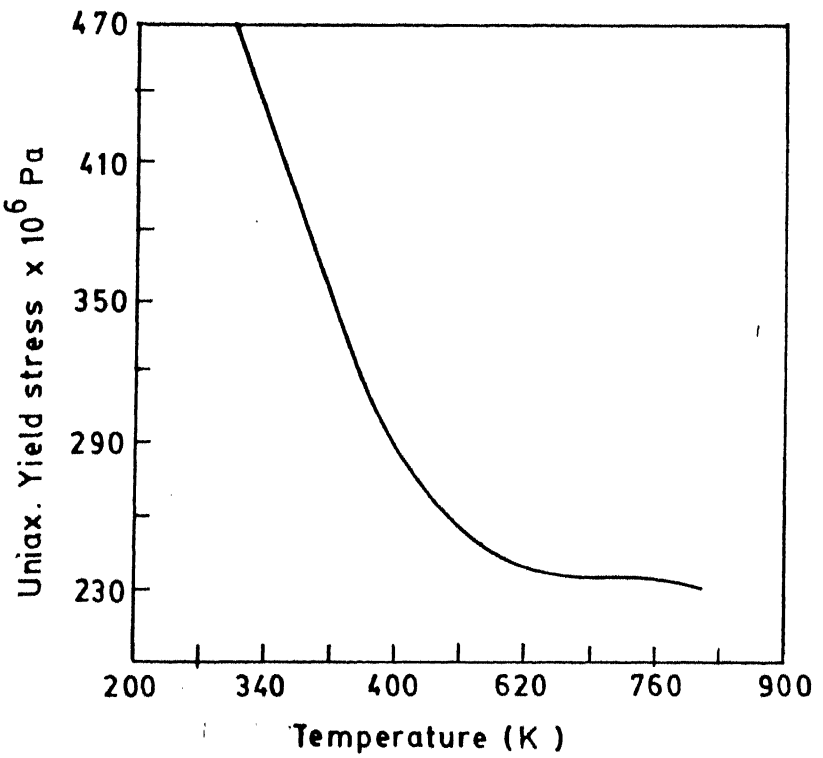


Fig. 2.6 Variation of Uniax Yield stress with temperature (for plain carbon steel)

minimum energy has been verified in the present analysis, by varying the chip thickness and finding the total plastic work and frictional work involved for each case.

2.7 Estimation of Material Properties

A careful study of the governing equation reveal that the material properties such as the uniaxial yield stress, thermal conductivity and specific heat affect the solution to a considerable extent. All the above mentioned properties vary with the temperature. In particular, at higher temperature the variations become quite significant.

In the present analysis, the solution domain is a minute region surrounding tool tip totally under plastic deformation. As already mentioned elsewhere, the temperatures in this region are quite high. Thus, to have a realistic picture of the results, the variation in these properties at higher temperatures is incorporated in the analysis.

The variation in thermal conductivity and specific heat of mild steel with respect to temperature is shown in Fig (2.4) and (2.7). This is taken from published data (30). The variation is uniaxial yield stress of mild steel with temperature is shown in Fig. (2.6) and has been taken from reference (28).

Metal cutting being a process involving high strain rates of the order of 10^3/S, the variation in yield stress is selected accordingly.

These variations are curve fitted and property correlations have been derived. For thermal conductivity and specific heat, piecewise equations of a straight line of the type

$y = mx + c$ have been fitted for different temperature ranges. For uniaxial yield stress, on the other hand a parabolic equation of the type

$$y = 9.6991 \text{ — } 2137677x + 1521.180\ x^2$$

has been derived using the least square method;, where

x = temperature in ^{o}K

and y = uniaxial yield stress in N/m^2.

CHAPTER 3

FINITE ELEMENT ANALYSIS

3.1 Introduction to Finite Element Technique

The finite element technique, whose engineering birth and boom in the 1960s was due to the application of digital computers to structural analysis, has spread to a variety of engineering and physical science disciplines in the last decade. In many engineering problems, it is always not possible to obtain a closed form exact solution. Thus, for such difficult problems, it is possible to get the best approximate solution (i.e. very close to exact one), using the powerful numerical techniques such as finite element method.

The basic concept of the finite element method is one of discretization. The finite element model is constructed in the following manner. A number of finite points are identified in the domain of the function, and the values of the function and its appropriate derivatives these points are selected as unknown variables. The points are called as nodes. The domain of the function is represented approximately by a finite collection of subdomains called finite elements. The domain is then an assemblage of elements connected together appropriately on their boundaries. The function is approximated locally within each element by continuous functions that are uniquely described in terms of nodal point values associated with the particular element. The path to the solution of a finite element problem consists of five specific steps :

(i) identification of the problem

(ii) discretization and definition of the element

(iii) establishment of the nodal equations for one element

(iv) assemblage of elemental contributions to nodal equations and incorporating boundary conditions and

(v) the numerical solution of the global equations.

The formation of element equations is done in four ways. (1) direct approach. (2) variational method (3) method of weighted residuals and (4) energy balance approach.

In the present analysis, Galerkin's weighted residual method has been used.

3.2 APPLICATION OF FEM

3.2.1 Grid Generation :

The first step in the finite element method is the selection of solution domain and its discretization. The choice of the solution domain boundaries for the metal cutting problem was discussed in previous chapter (see, Fig. 2.2). Grid generation for such a domain is quite a tedious task.

Moreover, it can be seen that the variation in depth of cut, chip thickness, rake angle or contact length leads to a totally new mesh. Thus, it is obvious that, unless one has an efficient way of grid generation, the analysis can not be done conveniently. Thus, grid generation forms one of the important aspects of the present analysis.

Considering the unsymmetry of the domain, it is divided into two zones, namely zone 1 and zone 2. The shapes of these zones are defined as shown in the Fig. (3.1).

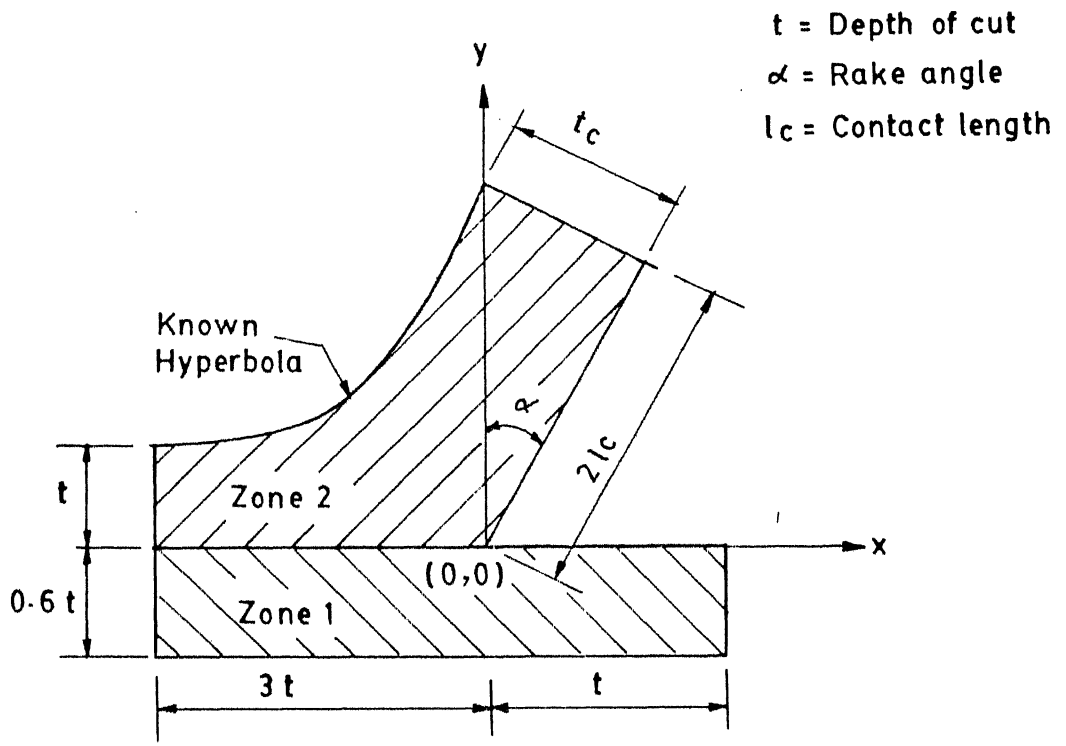


Fig. 3·1 Division and size of the solution domain set for grid generation

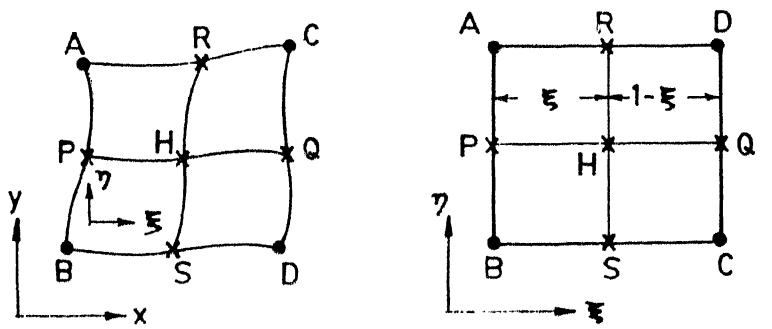


Fig. 3·2 Concept of trans-finite interpolations

Coordinate data and nodal connectivity of elements for each zone are calculated separately first and global elemental and nodal numbers for both zones put together are assigned later. To begin with, the coordinates for all the boundary nodes are prescribed suitably and then using a special type of interpolation, known as trans-finite interpolation technique, coordinate data of interior points have been generated. This technique is based on transforming an arbitrary 4-sided domain into a square domain in terms of body-fitting coordinates (ξ, η). The Cartesian and body fitting coordinates are shown in Fig. 3.2.

The procedure for generation x and y values for a typical point H is illustrated in Fig. 3.2. The domain ABCD is mapped into a unit square (1x1) in the transformed (ξ-η) domain. If H is the intersection point of the grid lines PQ and RS (which are η = const. and ξ = const. lines respectively), by bi-directional interpolation the x and y coordinates of H can be obtained in terms of the x and y coordinates of the boundary points P,Q,R and S. For instance, if ξ and 1-ξ are the distances of H from P and Q respectively in the transformed domain, then for one-directional interpolation

$$x_H = (1-\xi)\, x_P + \xi\, x_Q \qquad (3.1a)$$

$$y_H = (1-\xi)\, y_P + \xi\, x_P \qquad (3.1b)$$

However, for two directional interpolations the interpolations must be applied in both ξ and η directions suitably and (x_H, y_H) are then obtained in terms of the coordinates of the boundary points P,Q,R and S. Similarly for obtaining the coordinates of all the interior points, the appropriate boundary points (Fig.

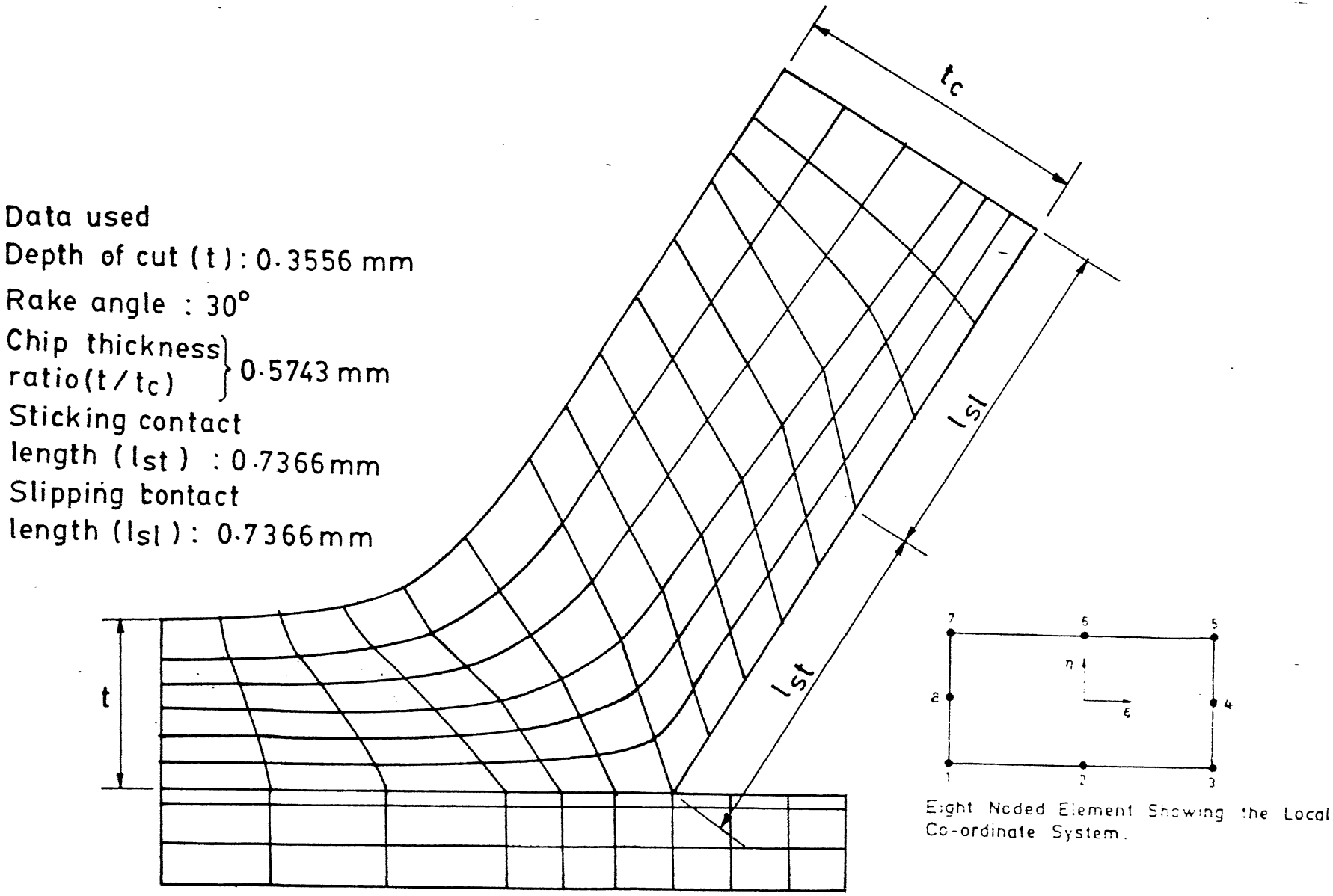


Fig. 3.3 A sample of the mesh generated by the grid generation software

3.1) as 4-sided, by transfinite interpolation the interior coordinates can be generated. A typical mesh thus generated is shown in Fig. (3.3).

In the present analysis, a grid with 386 nodes and 111 elements has been used. It was observed that, lesser number of elements and nodes and large sized elements, often led to the numerical difficulties such as negative determinant value etc. Provision has been kept in the program, to change the number of elements and the size of the elements to overcome such problems.

3.2.2. DERIVATION OF FINITE ELEMENT EQUATIONS

The finite element equations for plastic flow of metal, are better derived from the basic vector equations so that the handling of the traction boundary conditions at the chip-tool interface could be easily implemented.

FLOW EQUATIONS

The vectors momentum equation is given by

$$\rho(\underset{\sim}{V}.\underset{\sim}{\nabla}\underset{\sim}{V}) - \underset{\sim}{\nabla}.\ \underset{\approx}{\tau} - \underset{\sim}{f} = [0] \qquad (3.2)$$

By Galerkin technique, the vector residue equation for two dimensional cutting can be formed as

$$\iint Ni \left\{\rho\underset{\sim}{V}.\underset{\sim}{\nabla}\underset{\sim}{V}) - \underset{\sim}{\nabla}.\ \underset{\approx}{\tau} - \underset{\sim}{f}\right\} dx\ dy = [0] \qquad (3.3)$$

The term $\underset{\sim}{f}$ in equation (3.2) can be included in the pressure (isotropic part of stress) by redefining pressure as

$$\underset{\sim}{\nabla}p' = \underset{\sim}{\nabla}p + \underset{\sim}{f}$$

The term $\iint Ni\ (\underset{\sim}{\nabla}.\ \underset{\approx}{\tau})\ dx\ dy$ can be written by the weak formulation

as

$$\iint \mathrm{Ni}\ (\underset{\sim}{\nabla}.\ \underset{\approx}{\sigma})\ dx\ dy = \iint_D \left[\underset{\sim}{\nabla}\ (\mathrm{Ni}\ \underset{\approx}{\sigma}) - \underset{\sim}{\nabla}\ \mathrm{Ni}.\ \underset{\approx}{\sigma}\right] dx\ dy \qquad (3.4)$$

Expanding the first term in parenthesis by divergence theorem, the above equation (3.4) reduces to

$$\iint \mathrm{Ni}\ (\underset{\sim}{\nabla}.\ \underset{\approx}{\sigma})\ dx\ dy = \oint_B \mathrm{Ni}\ \underset{\sim}{n}.\underset{\approx}{\sigma}\ dl - \iint \underset{\sim}{\nabla}\ \mathrm{Ni}.\ \underset{\approx}{\sigma}\ dx\ dy \qquad (3.5)$$

where the first terms gives the contributions due to the specified traction boundary conditions. The original momentum residue equation (3.2) can now be rewritten as

$$\iint \mathrm{Ni}\ \rho\ (\underset{\sim}{V}.\underset{\sim}{\nabla}.)\underset{\sim}{V}\ dx\ dy + \iint \underset{\sim}{\nabla}\ \mathrm{Ni}.\ \underset{\approx}{\sigma}\ dx\ dy = \oint_B \mathrm{Ni}\ \underset{\sim}{n}.\underset{\approx}{\sigma}\ dl \qquad (3.6)$$

Expanding $\underset{\approx}{\sigma}$ into its components, the LHS term $\underset{\sim}{\nabla}\ \mathrm{Ni}.\underset{\approx}{\sigma}$ can be written as

$$\underset{\sim}{\nabla}\ \mathrm{Ni}.\underset{\approx}{\sigma} = \left\{\frac{\partial \mathrm{Ni}}{\partial x}\ \sigma_{xx} + \frac{\partial \mathrm{Ni}}{\partial x}\ \sigma_{yx}\right\} \hat{i} + \left\{\frac{\partial \mathrm{Ni}}{\partial x}\ \sigma_{xy} + \frac{\partial \mathrm{Ni}}{\partial x}\ \sigma_{yy}\right\} \hat{j} \qquad (3.7)$$

Momentum equations :

Putting the above term of equation (3.7) into the non-dimensional form of momentum equation (3.6), the x component equation can be derived as

$$\rho V^2 t \iint \mathrm{Ni} \left\{u^* \frac{\partial u^*}{\partial x^*} + v^* \frac{\partial u^*}{\partial x^*}\right\} dx^* dy^*$$

$$+\sigma_{yo} t \left\{\iint \frac{\partial \mathrm{Ni}}{\partial x^*}\left[-p^* + 2\mu^* \frac{\partial u^*}{\partial x^*}\right] dx^* dy^*\right\}$$

$$+\sigma_{yo} t \left\{\iint \frac{\partial \mathrm{Ni}}{\partial y^*}\left[\mu^*\left(\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial x^*}\right)\right] dx^* dy^*\right\} = \sigma_{yo} t \oint \mathrm{Ni}(\underset{\sim}{n}.\underset{\approx}{\sigma}^*.\hat{i})\ dl \qquad (3.8)$$

Similarly y-momentum equation can be obtained by taking the dot product between unit vector $\hat{j}$ and vector momentum euqation (3.6).

In the above equation, the inertial terms have been quasi-linearised by using the velocities corresponding to previous iteration (or initial guess) values during the iterative solution of the problem.

The field variables have been interpolated within each element as shown below

$$u = \sum_{i=1}^{n} Niui \quad , \quad v = \sum_{i=1}^{n} Nivi$$
$$p = \sum_{l=1}^{m} N_l P_l \quad , \quad T = \sum_{l=1}^{n} NiTi \qquad (3.9)$$

Similarly, the spatial coordinates of the boundaries of the isoparametric element are defined as

$$x = \sum_{i=1}^{n} Nixi \quad , \quad y = \sum_{i=1}^{n} Niyi \qquad (3.11)$$

where, n = 1 to 8

m = 1 to 4.

It may be noted that the pressure is defined only at corner nodes of the element and hence needs only a linear interpolation function.

All other variables (u,v and T) have been interpolated using quadratic shape functions.

For easy computation, the interpolation function Ni are defined in terms of local coordinates as under.

For corner noes ;

$$Ni = \frac{1}{4}\,(1+\xi_i\xi)\,(\xi_i\xi+\eta_i\eta-1)\,(1+\eta_i\eta) \qquad (3.12)$$

For mid-side nodes;

$$\left.\begin{aligned} Ni &= \tfrac{1}{2}\,(1-\xi^2)\,(1+\eta_i\eta),\ \xi_i = 0 \\ Ni &= \tfrac{1}{2}\,(1+\xi_i\xi)\,(1-\eta^2),\ \eta_i = 0 \end{aligned}\right\} \qquad (3.13)$$

Substituting the expressions for the interpolation of u,v and p into the momentum equation (3.8) and rearranging, we get the form of non-dimensionalized x-momentum equation as :

$$\left[\iint\left\{\frac{Ni}{\sigma}\left(N_k u_k^* \frac{\partial Nj}{\partial x^*} + N_k v_k^* \frac{\partial Nj}{\partial y^*}\right) + 2\mu^* \frac{\partial Ni}{\partial x^*}\frac{\partial Nj}{\partial x^*} + \mu^* \frac{\partial Ni}{\partial y^*}\frac{\partial Nj}{\partial y^*}\right\} dx^* dy^*\right][u_j^*]$$

$$+\left[\iint\left\{\mu^* \frac{\partial Ni}{\partial y^*}\frac{\partial Nj}{\partial y^*}\right\} dx^* dy^*\right][v_j^*]$$

$$-\left[\iint\left\{\left(\frac{\partial Ni}{\partial x^*} M_j\right) dx^* dy^*\right][p_j^*] = \left[\oint Nit_x^*\, dl^*\right] \qquad (3.14)$$

where $\sigma = \sigma_{yo}/\rho V^2$

t_x^* = Traction in x-direction (non-dimensionalized.)

$= t_x/\sigma_{yo}$

Similarly y-momentum equation can be obtained in the form

$$\left[\iint\left\{ \mu^{*} \frac{\partial Ni}{\partial x^{*}} \frac{\partial Nj}{\partial x^{*}}\right\} dx^{*}dy^{*}\right][u_j^{*}]$$

$$+\left[\iint\left\{\frac{Ni}{\sigma}(N_k u_k^{*}\frac{\partial Nj}{\partial x^{*}} + N_k v_k^{*}\frac{\partial Nj}{\partial y^{*}}) + \mu^{*} \frac{\partial Ni}{\partial x^{*}} \frac{\partial Nj}{\partial x^{*}}\right.\right.$$

$$\left.\left.+2\mu^{*} \frac{\partial Ni}{\partial y^{*}} \frac{\partial Nj}{\partial y^{*}}\right\} dx^{*}dy^{*}\right][v_j^{*}]$$

$$-\left[\iint\left\{(\frac{\partial Ni}{\partial y^{*}}M_j)dx^{*}dy^{*}\right][p_j^{*}] = \left[\oint Nit_y^{*}dl^{*}\right] \qquad (3.15)$$

where t_x and t_y are the traction components per unit area, given by,

$$t_x = \underset{\sim}{n}.\ \underset{\sim}{\sigma}.\ \hat{i}$$

and $$t_y = \underset{\sim}{n}.\ \underset{\sim}{\sigma}.\ \hat{j}$$

The residue minimization principle for the mass balance equation takes the form

$$\iint_D M_i \left\{ \frac{\partial u^{*}}{\partial x^{*}} + \frac{\partial v^{*}}{\partial y^{*}}\right\} dx^{*}dy^{*} = 0 \qquad (3.16)$$

The weighting functions for the mass balance equation have been chosen to be the linear shape functions used to interpolate pressure within the element. This is done in order to get a pressure field that is compatible with the incompressibility condition. Substituting the interpolation expression for the velocity variables, the final form of the mass balance equation is

$$\left[\iint M_i^{*} \frac{\partial Nj}{\partial x^{*}} dx^{*}dy^{*}\right][u_j^{*}] + \left[\iint M_i^{*} \frac{\partial Nj}{\partial y^{*}} dx^{*}dy^{*}\right][v_j^{*}] = 0 \qquad (3.17)$$

ENERGY EQUATION

The weighted residue form of the energy equation is

$$\iint Ni \left\{ \frac{\partial}{\partial x^*} \left(k^* \frac{\partial T^*}{\partial x^*} \right) + \frac{\partial}{\partial y^*} \left(k^* \frac{\partial T^*}{\partial y^*} \right) \right\} dx^* dy^*$$

$$+ \iint Ni \; Pe \; \frac{\sigma^*}{\sqrt{3}} \dot{\bar{e}}^* dx^* dy^*$$

$$- \iint NiPe \left\{ C_P^* \, u^* \frac{\partial T^*}{\partial x^*} + C_P^* \, v^* \frac{\partial T^*}{\partial y^*} \right\} dx^* dy^* = 0 \qquad (3.18)$$

Applying Green's theorem to 1st term of the above equation, the final form of energy residue equation is obtained as,

$$\left[\iint \left\{ k^* \frac{\partial Ni}{\partial x^*} \frac{\partial Nj}{\partial x^*} + k^* \frac{\partial Ni}{\partial y^*} \frac{\partial Nj}{\partial y^*} \right\} + NiP_e \left\{ C_P^* (N_k u_k^* \frac{\partial Nj}{\partial x^*} + N_k v_k^* \frac{\partial Nj}{\partial y^*}) \right\} \right]$$

$$dx^* dy^* \Big] [T_j^*]$$

$$= \oint Ni \; k^* \frac{\partial T^*}{\partial n^*} \, d\ell^* + \iint Ni \; P_e \frac{\sigma^*}{\sqrt{3}} \dot{\bar{e}}^* dx^* dy^* \qquad (3.19)$$

The second term on the RHS is the heat generation contribution and the first term is from heat transfer boundary conditions.

3.2.3 BOUNDARY CONDITIONS

Modelling of boundary conditions is as important as the modelling of differential equation themselves. In fact, experience reveals that the toughest part of problem formulation is the prescription of proper boundary conditions.

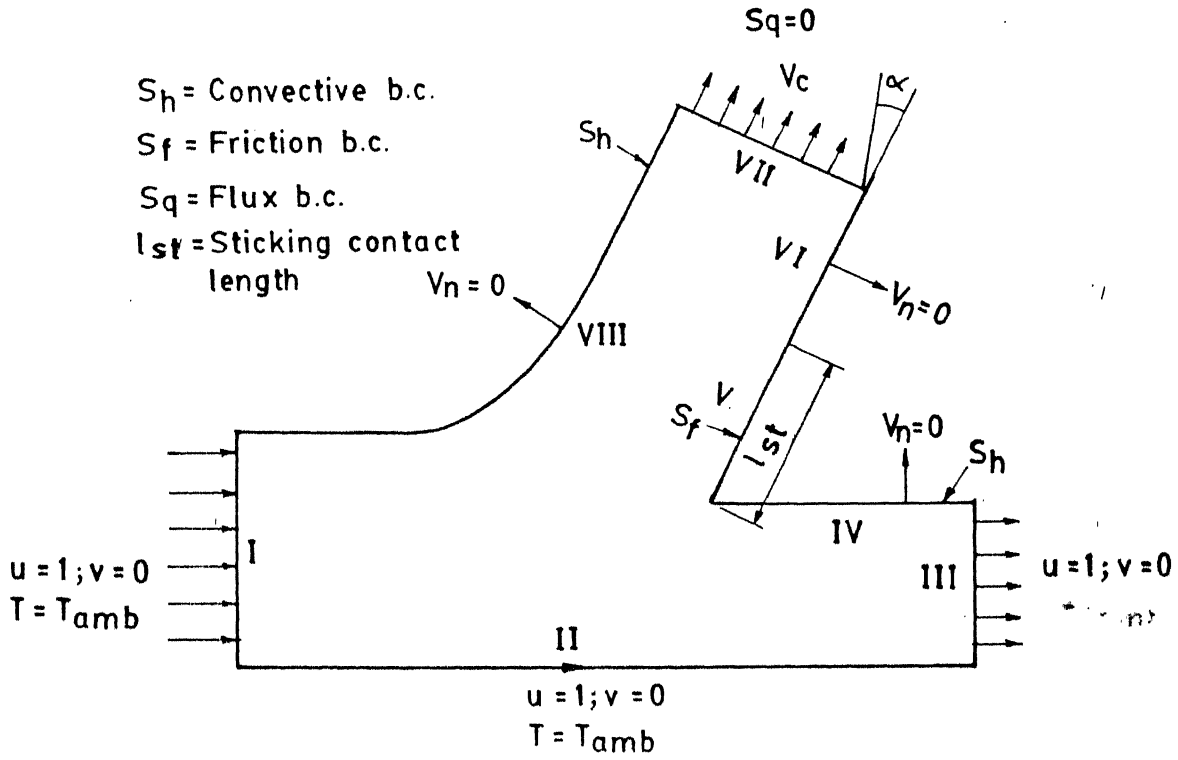


Fig. 3.4 Problem regions showing the boundary conditions

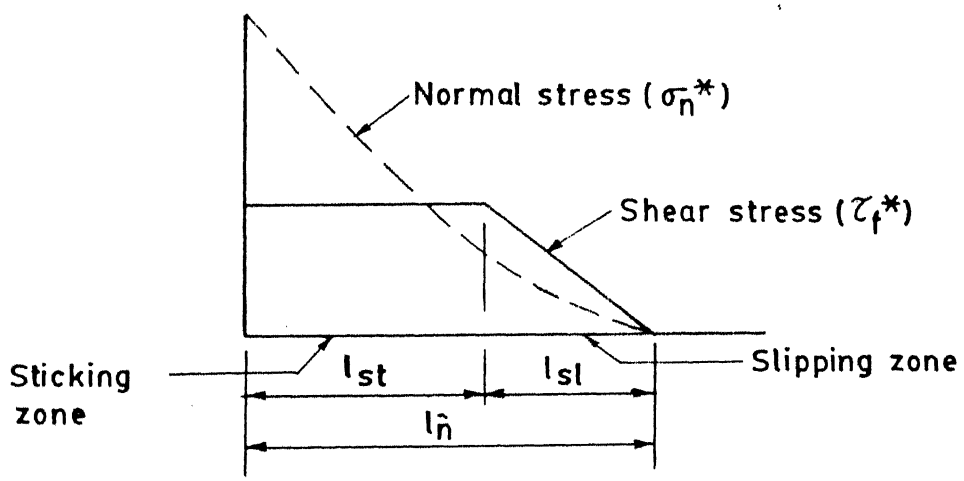


Fig. 3.5 Idealized distributions of normal and shear stress on the tool rake face

The facilitate computation and to implement boundary conditions in a systematic manner, the total boundary has been divided in to eight separate segments as shown in Fig. (3.4). Surfaces I, II, and III have prescribed boundary conditions for u,v and T. These conditions are implemented by incorporating them in nodal equations for corresponding variables, in the final matrix equations. Also, the first node of the mesh is prescribed with a pressure boundary condition.

The normal velocity ($\underset{\sim}{V}_n = 0$) is applied on surfaces IV, V, VI and VIII, since there is no flow in a direction normal to these surfaces. Thus, normal velocity at the surface requires.

$$\underset{\sim}{V}^*_n = u^* \cos\theta\, \hat{i} - v^* \sin\theta\, \hat{j} = 0 \qquad (3.21)$$

where θ is the angle made by the normal with the x-axis and $(\hat{i}, \hat{j})$ are unit vectors in (x,y) direction respectively.

Expanding u and v in equation (3.21),

$$\underset{\sim}{V}^*_n = \sum_{j=1}^{8} N_j u^*_j \cos\theta\, \hat{i} - \sum_{j=1}^{8} N_j v^*_j \cos\theta\, \hat{j} = 0 \qquad (3.22)$$

where u^*_j and V^*_j are the nodal velocities of the element which lies on the concerned boundary.

On surface VII, which is the far end of the chip, the velocity is equal to the chip velocity V^*_c. The velocity components u and v on this boundary are given by

$$\left.\begin{aligned} u^* &= V^*_c \sin\alpha \\ v^* &= V^*_c \cos\alpha \end{aligned}\right\} \qquad (3.23)$$

where, α = Rake angle

On the surface V and VI, we apply two boundary conditions.

i) zero normal velocity ie $V_n = 0$

ii) traction boundary condition

For the normal direction, zero normal velocity condition is implemented.

The traction boundary condition for shear stress can be applied by considering the momentum equation in the tangential direction as given below :

(x-momentum equation) $(-\sin\alpha)$ + (y momentum equation) $(-\cos\alpha)$

$$= \int Ni\ \tau_F^*\ d_l^*$$

where, τ_F^* = shear stress along the surface ($= m\mu^*\dot{\bar{e}}^*$)

i.e.
$$-\left[\iint\left\{\frac{Ni}{\sigma}\left(N_k u_k^* \frac{\partial Nj}{\partial x^*} + N_k v_k^* \frac{\partial Nj}{\partial y^*}\right) + 2\mu^* \frac{\partial Ni}{\partial x^*} \frac{\partial Nj}{\partial x^*} + \mu^* \frac{\partial Ni}{\partial y^*} \frac{\partial Nj}{\partial y^*}\right\} dx^* dy^*\right] \sin\alpha\ [u_j^*]$$

$$-\left[\iint\left\{\mu^* \frac{\partial Ni}{\partial y^*} \frac{\partial Nj}{\partial x^*}\right\} dx^* dy^*\right] \sin\alpha\ [v_j^*]$$

$$+\left[\iint\left(\frac{\partial Ni}{\partial x^*} M_j\right) dx^* dy^*\right] \sin\alpha\ [P_j^*]$$

$$-\left[\iint\left\{\mu^* \frac{\partial Ni}{\partial x^*} \frac{\partial Nj}{\partial y^*}\right\} dx^* dy^*\right] \cos\alpha\ [u_j^*]$$

$$-\left[\iint\left\{\frac{Ni}{\sigma}\left(N_k u_k^* \frac{\partial Nj}{\partial x^*} + N_k v_k^* \frac{\partial Nj}{\partial y^*}\right) + \mu^* \frac{\partial Ni}{\partial x^*} \frac{\partial Nj}{\partial x^*} + 2\mu^* \frac{\partial Ni}{\partial y^*} \frac{\partial Nj}{\partial y^*}\right\} dx^* dy^*\right] \cos\alpha\ [v_j^*]$$

$$+\left[\iint\left(\frac{\partial Ni}{\partial y^*} M_j\right) dx^* dy^*\right] \cos\alpha\ [P_j^*] = \int Ni\ \tau_F^* d_l^* \qquad (3.24)$$

CENTRAL LIBRARY
Acc. No. 113088

The traction boundary condition is extended to the elements on VIth surface also, with the assumption that shear stress decreases linearly towards the end of the contact patch and becomes zero. This based on the observation of many investigators (19) that shear stress remains reasonably constant on the sticking contact length and reaches zero gradually in the slipping contact length.

For surfaces IV and VIII which have convective heat transfer boundary condition, the overall heat transfer coefficient is given as input data.

The weighted residual form of the boundary condition is

$$\int Nik^* \frac{\partial T^*}{\partial n^*} dl = -\int Nih^* (T^* - T^*_{amb})\, dl = -\int Nih^*\, T^* dl + \int Nih^*\, T^*_{amb} dl \qquad (3.25)$$

Expanding $T = \sum_{j=1}^{8} N_j T_j$, convective heat loss is given by,

$$\int Nik^* \frac{\partial T^*}{\partial n^*} dl = \left[\int (Nih^*\, N_j)\, dl\right] [\, T_j\,] - \left[\int Nih^*\, T^*_{amb} dl\, \right] \qquad (3.26)$$

On the chip tool interface (surfaces V and VI), the condition for frictional heat generation is considered

This is given by

$$-k^* \frac{\partial T^*}{\partial n^*} = \phi_k\, P_e\, \tau_F^*\, V_T^*$$

or

$$\frac{\partial T^*}{\partial n^*} = -\frac{1}{k^*} \left[\phi_k\, P_e\, \tau_F^*\, V_T^*\right] \qquad (3.27)$$

weighted residual form of the above equation is

$$\int_V Ni \frac{\partial T^*}{\partial n^*} \, d\ell = \phi_k P_e \int \frac{Ni}{k^*} \tau_F^* V_T^* d\ell \qquad (3.28)$$

The slip velocity V_T is given by,

$$V_T^* = v^* \cos \alpha + u^* \sin \alpha \qquad (3.29)$$

ELEMENT ASSEMBLY :

In the previous section, the elemental contributions to the left hand side coefficient matrix and right hand side vector have been discussed in detail for the eight-noded isoparametric quadrilateral elements. These elemental contributions are assembled in to a global matrix equation by adding the entries corresponding to each nodal variable appropriately.

The boundary integral contributions on both left hand side and right hand side of the matrix equation are also incorporated as discussed in the previous section. This assembly procedure results in a matrix equation of the form

$$[A]\,[x] = [B]$$

where [A] = coefficient matrix

[x] = solution vector

[B] = Right hand side vector

The problem at this stage is set for solving above matrix equation by any standard technique available.

3.2.4 Matrix Solution Technique :

For the present problem an iterative technique is used for matrix solution based on the Frontal method. Frontal method offers many advantages such as small use of memory space and high speed of computation. It utilizes the fact that when all the contributions for a particular node and for a particular variable are over during element assembly, it could be transferred to the disk memory. On this basis it retains in its memory only those variables which are yet to be assembled the variables are solved by Gaussian elimination is after assembly. Since the governing equations are highly non-linear an iterative procedure with successive under-relaxation has been employed.

3.3 Post processing

After solving the governing equations, the field variables, namely velocities, pressures and temperatures would be available at each node of the solution domain.

These variables can be further processed to get many interesting quantities such as forces, stress distribution etc. Also, it is possible to plot isotherms to study the temperature - distribution or iso-strain rate lines which might give an idea regarding the size of the deformation zone.

Thus, in the present analysis, the following post-processing has been done.

i) plotting of isotherms

ii) plotting of iso strainrate lines

and iii) estimation of forces involved in metal cutting

. **Isotherms :** Temperature has been predicted for all the nodes in the domain. Using linear interpolation, it is possible to estimate temperature at any point across the element boundary. The coordinates of points of with same temperature are calculated using this procedure and joined using smooth curves. At a simple glance, isotherms give a clear idea about the temperature distribution in the domain.

. **Isostrainrate Curves :** Once the velocity fields all over the domain are available, the strain rates can be evaluated very easily using equation (2.14). The strainrate plots give a good idea about the deformation pattern in the domain. The strain rates at different points are first normalized with respect to the maximum strain rate occurring in the solution domain. Points having the same value of normalized strain rates are connected by smooth curves and the curves 2%, 3%, 4%, 5%, 6%, 8% etc. are thus plotted. These constant strain rates provide a clear picture of primary and secondary deformation zones.

. **Estimation of Forces :** Forces acting on the tool are of paramount importance. Now, once the velocity fields, pressure fields etc. are available in the domain, finding stresses and forces is not difficult. For force calculation, only the elements lying on the rake surface would come into the picture. (i.e. surface V and VI). For there elements the stresses can be easily calculated using the following relations :

$$\sigma^{*}_{xx} = -p^{*} + 2\mu^{*} \frac{\partial u^{*}}{\partial x^{*}}$$

$$\sigma^{*}_{yy} = -p^{*} + 2\mu^{*} \frac{\partial v^{*}}{\partial x^{*}}$$

and $$\sigma^{*}_{xy} = \sigma^{*}_{yx} + \mu^{*} \left[\frac{\partial u^{*}}{\partial y^{*}} + \frac{\partial v^{*}}{\partial x^{*}} \right]$$

Further, these stresses can be used to find the shear stress and normal stress on the rake surface, using,

$$\sigma_{nn}{}^{*} = \frac{(\sigma^{*}_{xx} + \sigma^{*}_{yy})}{2} + \frac{(\sigma^{*}_{xx} - \sigma^{*}_{yy})}{2} \cos 2\alpha + \sigma^{*}_{xy} \sin 2\alpha$$

and

$$\tau^{*}_{F} = \frac{-(\sigma^{*}_{xx} - \sigma^{*}_{yy})}{2} \sin 2\alpha + \sigma^{*}_{xy} \cos 2\alpha$$

τ^{*}_{F} also can be calculated using the relation, $\tau^{*}_{F} = \mu^{*}\, m\, \dot{\bar{e}}^{*}$

where, m = friction factor

$\dot{\bar{e}}$ = stranrate [effective]

σ^{*}_{nn} = Normal stress acting on the rake surface

τ^{*}_{F} = shear stress acting on the rake surface.

From these, we can easily calculate normal force and shear force acting on the rake surface.

$$F^{*}_{n} = \int_{(dl)} \sigma^{*}_{nn} \times \text{contact length} \times \text{width}$$

$$F^{*}_{s} = \int_{(dl)} \tau^{*}_{F} \times \text{contact length} \times \text{width.}$$

where, F^{*}_{n} = normal force

F_s^* = shear force

F_n^* and F_s^* can be easily resolved to get the desired forces viz F_x and F_y

$$F_x^* = - F_s^* \sin\gamma - F_n^* \cos\gamma$$

$$F_y^* = - F_n^* \sin\gamma - F_s^* \cos\gamma$$

where, F_x^* = cutting force or thrust force

F_y^* = feed force.

γ = rake angle

These forces can be expressed in dimensional form as,

$$F_x = F_x^* \times \sigma_{yo} \times t^2$$

$$F_y = F_y^* \times \sigma_{yo} \times t^2$$

where, σ_{yo} = uniaxial yield stress of the workpiece material at ambient temperature

t = Depth of cut.

Since, we are using iterative solution technique, for the calculation of forces, the fully converged values of the variables are used.

3.4 Programming

Computer programming is one of the challenging tasks of the present work.

The programming work is divided in two parts.

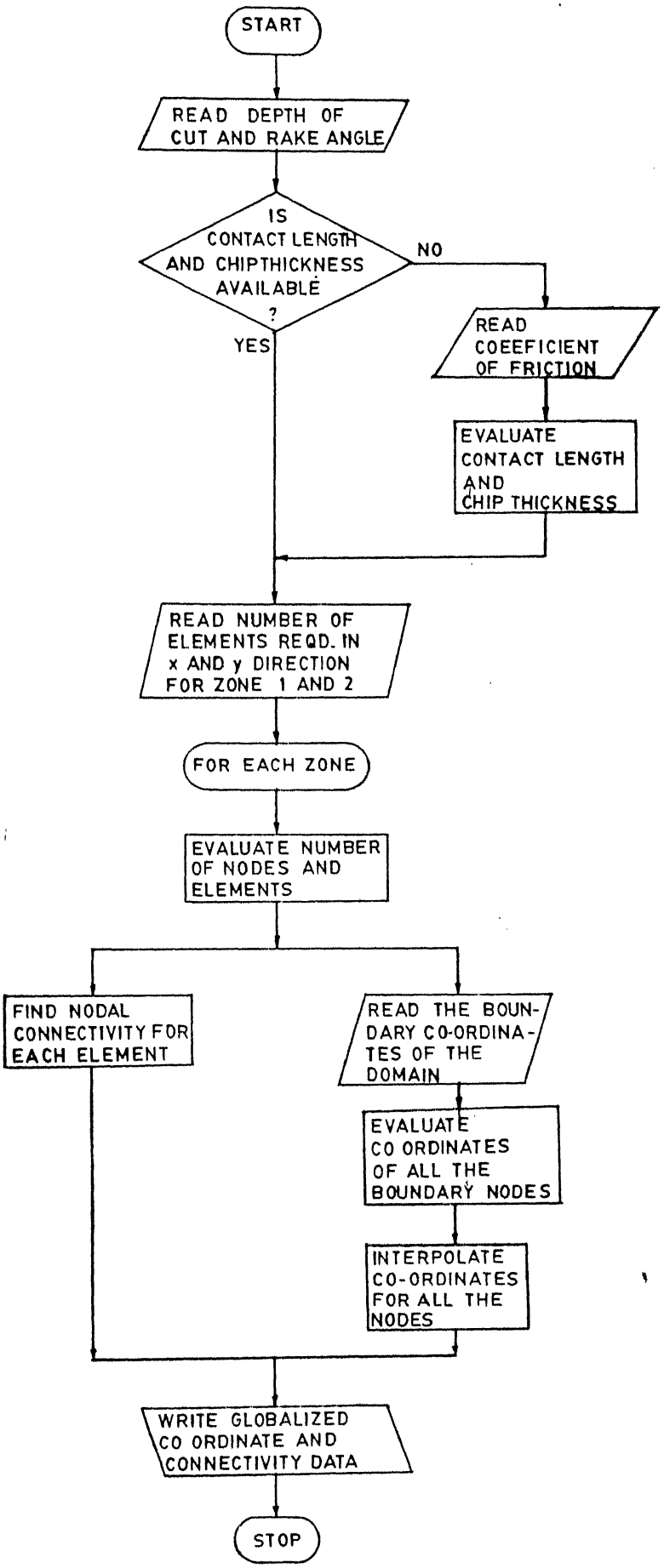


Fig.3.6 Flowchart for grid data generation

(i) program for grid generation.

(ii) program for finite element analysis

3.4.1 Program for grid generation :

A simple flowchart shown in Fig. (3.6) gives a general picture of the procedure involved.

In the main programme itself, the data related to the cutting process such as depth of cut, rake angle, contact length and chip thickness are read. If contact length and chip thickness are not available, provision has been kept to calculate them within the program. All the distances have been non-dimensionalized using the depth of cut. Other relevant data for the domain such as number elements in x direction, and y-direction (for each zone), number of common elements, number nodes per element etc. is also read in the MAIN itself.

Subroutine ZONE is called individually for each zone to generate nodal connectivity and them a globalised connectivity is generated using subroutine SUPER after considering common elements.

Subroutine AUTO is called to generate co-ordinate data for each node, separately for each zone and later this data is globalised.

Coordinates of the corner points of the zones and of some specific points on the boundary (eg. the point from which a finer mesh is desired etc.) is supplied in subroutine INPUT. Using this data, first, coordinates for all the boundary nodes is generated. Generation of these coordinates is simple so long as

the boundaries are straight.

For the eighth surface, where the concept of hyperbolic stream line is used (explained elsewhere). Subroutine CURVE is called, to generate the boundary nodal coordinates along this curved surface.

All this coordinate data for the nodes on the boundaries, is used to generate the coordinates for all the elements in the domain in the subroutine INTPOL, where the concept of trans-finite interpolation (explained elsewhere) is used. This is done for each zone separately. Finally, all these datas of individual zones are globalized.

3.4.2 Program for Finite Element Analysis

Fig. 3.7 gives the flow-chart of the program.

Main program calls the subroutine DIMENS, DINPUT, ITERAT and FORCE, which inturn call the other subroutines.

Subroutine DIMENS returns the values of the array dimensions which remain unchanged throughout the program.

Subroutine DINPUT reads and returns all the input data such as cutting conditions, initial conditions, boundary conditions, nodal coordinate data, element connectivity data and prescribed parameters like ambient temperature and material properties at ambient temperature. The whole input data is in the non-dimensionalized form. Some of the inputs are checked for correction by two routines named DIAGN1 and DIAGN2.

Subroutine ITERAT calls the FRONTS routine for solving assembled matrix equation. This routine begins the problem with

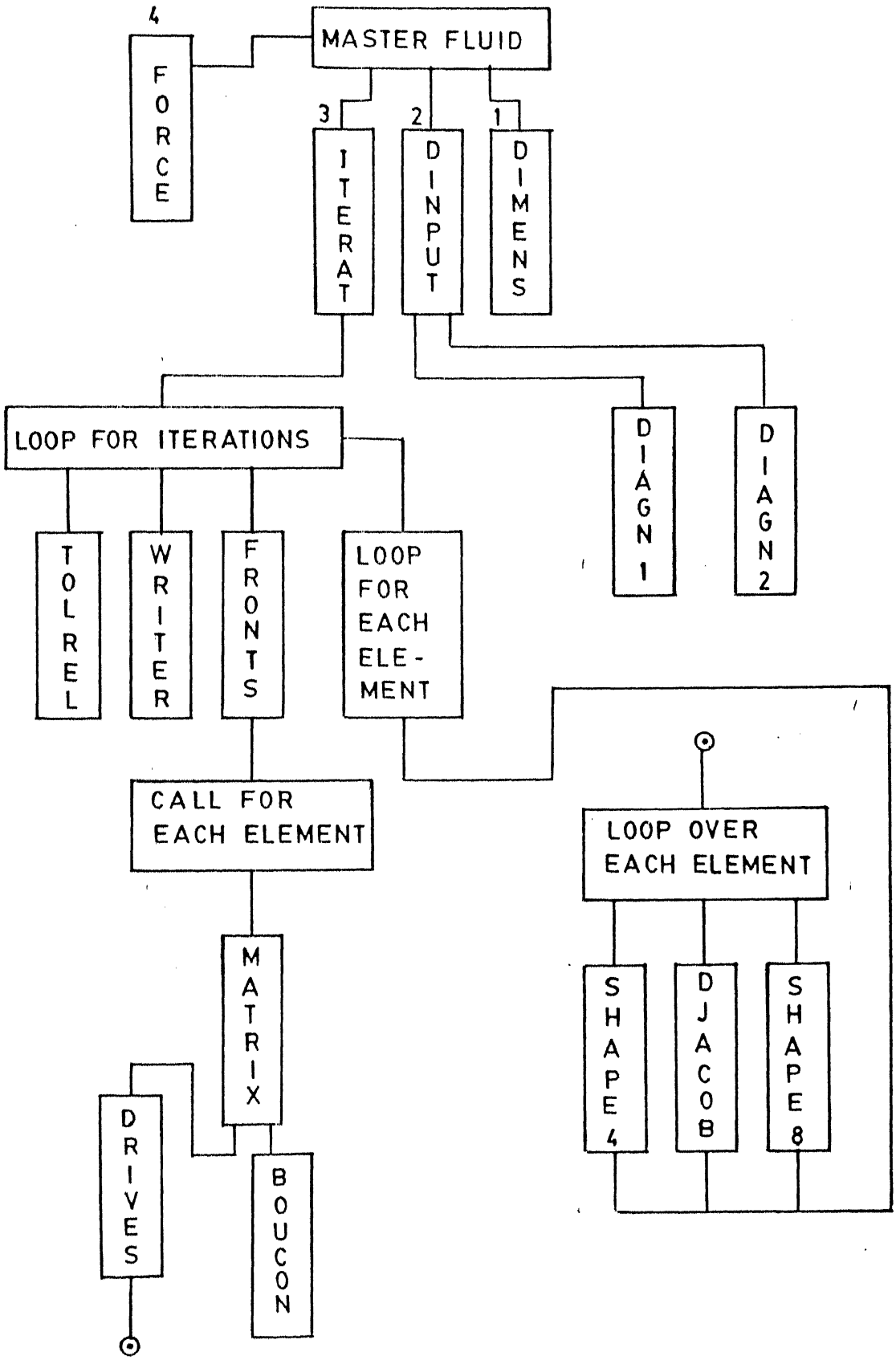


Fig. 3.7 Flowchart of Finite element analysis program

guessed initial values for all the nodal variables and relaxes them for next iteration in the subroutine TOLREL, if the solution is not found converged within the prescribed tolerance. This routine also calls the WRITER subroutine (i) to write the nodal values of the variables if the solution has converged and (ii) to write the largest relative change in the variables that has occurred during a particular iteration if the solution is not converged.

Assembling is MATRIX subroutine which calculates the element fluid matrix flumx. This fluid matrix consists of the elemental contributions of the governing equations to form coefficient matrix. Similarly elemental right hand side vector is also calculated here. Subroutine DRIVES is called within MATRIX, which in turn calls the subroutines SHAPE8, DJACOB & SHAPE4 to calculate the shape function and their derivatives.

The boundary conditions are implemented in BOUCON which is called in MATRIX before it returns to FRONTS. This is called for only boundary elements. Here, relevant boundary conditions from among the zero normal velocity, convective heat loss, the frictional heat flux of frictional shear stresses are identified appropriately and elemental contributions are evaluated. The contributions to right hand-side vector are calculated at local nodes and are assembled in the global vector both in MATRIX & BOUCON subroutines.

Subroutine FORCE does the post processing part of the nodal values predicted after convergence. Frictional force and the normal force on the rake surface are evaluated considering the elements lying on this surface and finally, the cutting force and the feed force are evaluated from them.

CHAPTER 4

EXPERIMENTAL ANALYSIS

When we speak of theoretically estimating the quantities of practical importance such as forces and temperatures, it is obviously needed to check these predictions with those obtained experimentally.

The following experiments have been conducted in the present work.

1. Measurement of average temperature along the chip-tool interface.
2. Measurement of Cutting Forces.

The tests have been designed for various cutting conditions to find the effect of cutting speed, rake, feed etc. on the temperature distribution and Forces.

4.1 TEMPERATURE MEASUREMENT

The determination of temperature distribution on the rake face and in the cutting tool is very important for estimation of tool wear and tool life. Several techniques have been used by researchers for accurate determination of temperature at different points on the tool. Some of these are

(i) Use of photosensitive paints.

(ii) Use of tool workpiece thermocouple.

(iii) Using tiny thermocouples which are embedded in the tool and

(iv) Using infrared photographic technique.

Most of these techniques except the tool-work thermocouple technique are not popular because of the low spread of response or complexity. Although tool-workpiece thermocouple procedure gives only the average temperature, it is a very simple and reliable method. This method has been employed here.

4.1.1 Tool-Workpiece Thermocouple Method

This method makes use of the emf produced between the hot contact patch of tool and workpiece and their cold ends. If the cold ends of the tool and workpiece are joined, a small current will flow which can be measured. Also, the emf generation can be measured in terms of volts.

This technique is based on the following laws of thermo-electric circuits (MCS).

1. The emf of thermoelectric circuit depends only on the difference in temperature between the hot and cold junctions, and is independent of the gradients in the parts making up the system.
2. The emf generated is independent of the size and resistance of the conductors.
3. If the junction of the two metals is at uniform temperature, the emf generated is not affected, if a third metal, which is also at the same temperature, is used to make the junction between first two.

A great number of thermocouple hot junctions are active at the chip-tool interface asperities which are forming a parallel electric network. Thus, the emf recorded would be an algebraic

average of all the thermal emfs. This average emf can be calibrated against temperature and the average temperature can be found by this technique.

4.1.2 Test Conditions

All the experiments have been carried out on a HMT lathe. The workpiece material selected was Mild steel. HSS tools of standard dimensions with following rake angles have been used.

10°, 15°, 20°

The tool was held firmly in the position of tool post within the tool holding arrangement provided on the dynamometer. A constant overhang of the tool of 18 mm was maintained for all the tests. The experiments were carried out under dry cutting conditions.

Due to some practical limitations of machine and the measuring devices, the experiments were conducted only for lower values of feed (e.g. capacity of loadcell $\longrightarrow$ 100 Kg.) (0.05 mm/rev to 0.075 mm/rev). The cutting speed was conveniently selected in the range 0.4 m/s to 0.8 m/s.

Rake angles of 10°, 15° and 20° were used. The following cutting conditions were maintained constant for all the tests : width of cut = 2.4 mm and clearance angle = 10°.

However, only a few of these readings taking care of the variations in all process parameters viz. feed, speed and rake etc. have been used for comparison.

4.1.3 Experimental Procedure

The block diagram of experimental set-up is shown in Fig. (4.1). The workpiece is insulated from the chuck using a insulating sheet. Even the tool holding arrangement within the dynamometer is insulated perfectly. Care has been taken that contact exist only between the tool and workpiece at the cutting edge. The workpiece and the tool tip constitute the hot junction H. The cold junctions A and B remain at room temperature. Mercury is used to make electrical contact with the rotating member in order to complete thermo-electric circuit. The wires C and D may be ordinary copper wires since both the ends are at the same temperature and these have no influence on the emf generated. The emf generated is recorded by a double channel recorder/ plotter. At the end of each experiment the tool and the workpiece were allowed to cool to the room temperature. This was done to ensure that the tool tip is always at room temperature at the beginning of the experiment. In each of the test cases, the emf was recorded after a stabilized trend was observed.

The measured values of emf were converted into temperature with the help of a calibration curve. The calibration was carried out by contacting the tool tip to a MS plate put in a furnace with sufficient insulation. The temperature of the hot junction was measured by a standard Chromel-Alumel thermocouple, which was later used for comparing with corresponding emf readings of tool- work thermocouple. Thus the calibration curve was prepared by varying the temperature of the MS plate in the furnace.

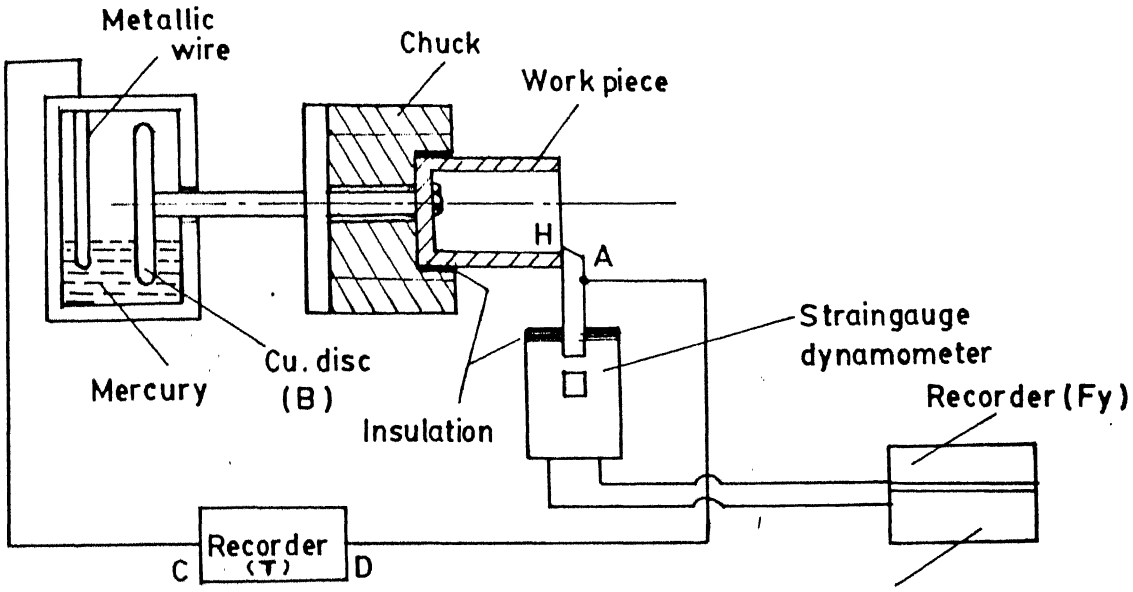


Fig. 4.1

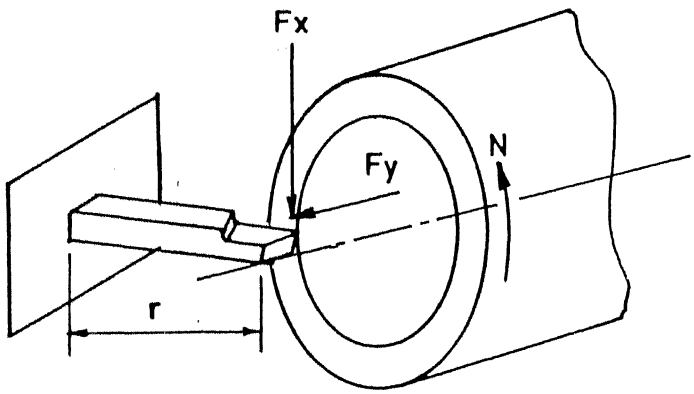


Fig. 4.2

Fig. 4.1 Exptl. setup for the measurement of average tool-chip interface temperature and forces.

Fig. 4.2 Schematic view of dynamometer for measuring cutting forces in two dimensional turning operation.

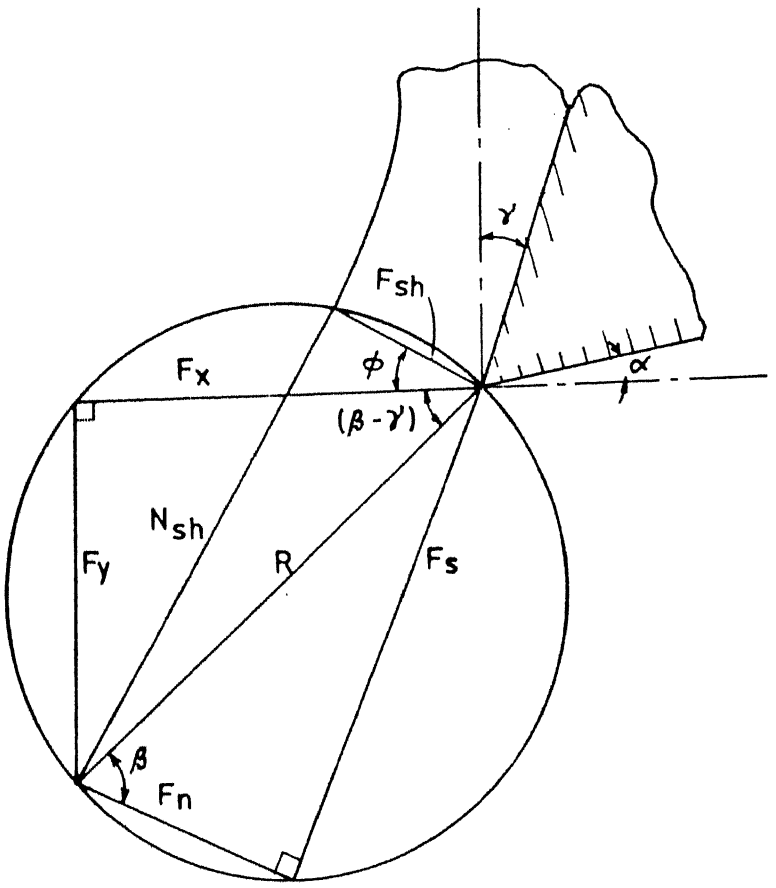


Fig. 4.3 Merchant's circle diagram

4.2 FORCE MEASUREMENT

4.2.1 Relationships between various force components :

The notations used for the forces are as follows.

i) Main cutting force coinciding with the principal cutting velocity direction is taken parallel to X-axis of the Cartesian coordinate system and is denoted as F_x (Fig. 4.2).

ii) Feed force acting in perpendicular direction to main cutting force in Y-direction, is denoted as F_y (Fig. 4.2).

The action of forces in a single point orthogonal two-dimensional cutting operation producing continuouschips can be readily understood from Merchant's circle diagram shown in Fig. (4.3). Here F_x and F_y are Cartesian force components acting upon the tool nose, normally called main or tangential component and radial or thrust component. The shear plane of the material sustains the normal (N_{sh}) and the shear component (F_{sh}). For force balance, from free body diagram of the chip we get the conditions that the resultant force (R) on the rake surface should be co-linear and equal in magnitude to that acting over the shear plane. Thus the circle with R as the diameter will contain all the force considered, and this diagram can be used conveniently for the mutual conversion of the component forces into equivalent shearplane and rake forces. Thus we can write,

$$F_{sh} = F_x \cos\phi - F_y \sin\phi$$

$$N_{sh} = F_y \cos\phi + F_x \sin\phi$$

$$F_s = F_x \sin\gamma + F_y \cos\gamma$$

$$F_n = F_x \cos\gamma - F_y \sin\gamma$$

If the friction coefficient on the rake face is μ, then we have

$$\beta = \tan^{-1} \mu$$

$$= \tan^{-1} \frac{F_s}{F_n}$$

where β is the friction angle.

In practice the external cartesian force components F_x and F_y are usually measured which is then made use of in further force evaluations using the equations already derived.

It is well known that force is an entity generally measured through its action on a system offering a finite resistance, which is usually pre-calibrated. Based upon the principles of modern transducer techniques, the measurement of cutting force hence involves, a transducer, signal conditioner which converts the transducer output into a useful measurable parameter and lastly an indicator or a recorder to register the final results.

The accurate measurement of F_x and F_y (components of resultant tool force) has been the subject of considerable effort in the past, and several types of cutting force dynamometers have been developed. In most of the metal cutting force dynamometers

the tool force is determined by measuring the deflections or strains in the elements supporting the cutting tool. It is essential that the instrument should have high rigidity and high natural frequencies so that the dimensional accuracy of cutting operation is maintained and tendency for chatter or vibrations to occur during cutting is minimized. They should have reliability over a range of temperatures.

4.2.2 Test Conditions for Force Measurements

Test conditions were maintained same as those used in temperature measurement.

The whole set of readings namely,

i) Force F_x

ii) Force F_y

iii) Temperature

iv) Chip thickness

were taken for various combinations of cutting conditions.

The dynamometer used in the experiment was cantilever type straingauge mounted. Concept of Wheatstone bridge was used in the circuit. The constructional details of the dynamometer is not being explained here, since it is easily available in any book on metal cutting. The capacity of the dynamometer was 100 Kg.

Omniscribe stripchart recorders were used for recording forces as well as temperature readings.

4.2.3 EXPERIMENTAL PROCEDURE

The experimental set up for orthogonal cutting is shown in Fig. (4.1). The workpiece selected was a mild steel seamless

tube of suitable dimensions. To begin with, the ID and OD of the tube were smoothened by taking some cuts, till thickness was same throughout.

After setting particular combination of cutting conditions, forces and temperature readings were taken simultaneously and some chips were collected. Chip thickness was found by using the weight method. For some of the readings, attempts were made to measure the contact length by applying lamp black on the rake surface and then, measuring length of scratch under a travelling microscope.

Many researchers have done a lot of experimentation of this type. We have used some of this work for comparing our results. Temperature distribution in workpiece and chip during orthogonal cutting was observed by Boothroyd [19] using infrared photograph technique. This has been used to compare the isotherm pattern predicted from our analysis. Muraka et al. [17] have also done a comprehensive experimental work consisting of measurement of forces in metal cutting. This has been used to compare the predicted forces from our analysis.

A summary of experimental results used for comparison is given in Table 1.

Table 1 : Cutting Conditions and Measured Data

Test No.	Workpiece Revs/min N rpm	Workpiece diameter D mm	Width of cut W mm	Speed V m/min	Feed t mm/rev	Rake Angle α (o)	Chip thickness ratio r (t/tc)	Shear angle ϕ (o)	Contact length * mm	Cutting force Fx N	Feed Force Fy N	Average Ch tool inter face temp. ^{o}K
1	57	162.10	2.35	29.02	0.05	10^{o}	0.49	27.8	0.14	563	389	479
2	57	162.10	2.35	29.02	0.063	10^{o}	0.484	27.4	0.17	641	416	484
3	57	162.10	2.35	29.02	0.075	10^{o}	0.5	28.3	0.2	702	452	502
4	71	162.10	2.35	36.15	0.075	10^{o}	0.49	27.8	0.2	667	453	500
5	90	162.10	2.35	45.83	0.075	10^{o}	0.46	26.2	0.2	633	449	502
+ 6	80	98.20	6.35	24.68	0.3556	20^{o}	0.536	39^{o}	1.11	3525	1560	
+ 7	80	98.55	6.35	24.77	0.3556	30^{o}	0.619	35^{o}	0.7366	2650	545	

Note :

. All the tests carried out in dry cutting conditions.
. For all the tests HSS single point cutting tool is used.
* Sticking contact length.
+ Experimental results of Muraka et al. [17].

CHAPTER 5

RESULTS AND DISCUSSIONS

In the present work, a finite element formulation for the thermal analysis of orthogonal metal cutting has been developed. Based on this formulation, a complete software package, including useful post and pre-processing modules has been provided.

The primary objectives of the present analysis have been as follows :

i) To predict velocity, pressure and temperature fields in vicinity of tool tip, by the finite element solution of coupled stress balance and heat balance equations.

ii) To conduct a detailed parametric study, to highlight the effects of process parameters such as, cutting velocity, chip thickness ratio, depth of cut etc.

iii) To predict quantities of practical importance such as forces involved in metal cutting, isothermal patterns and the shapes of deformation zones.

iv) To validate the theoretical model with experimental results.

v) To verify the application of energy minimization principle for determining the value of chip-thickness ratio.

In the course of analysis, some significant quantities such as strain rates, which are very important for the study of the deformation pattern in the plastic region, emerge out. The variation of strain rate distribution across the shear plane and along the shear plane have been studied to throw light, upon the

deformation processes undergone by the material. Also, the temperature distributions along the rake face in the contact region and in the vicinity of the shear plane have been plotted to analyse the thermal phenomena in detail.

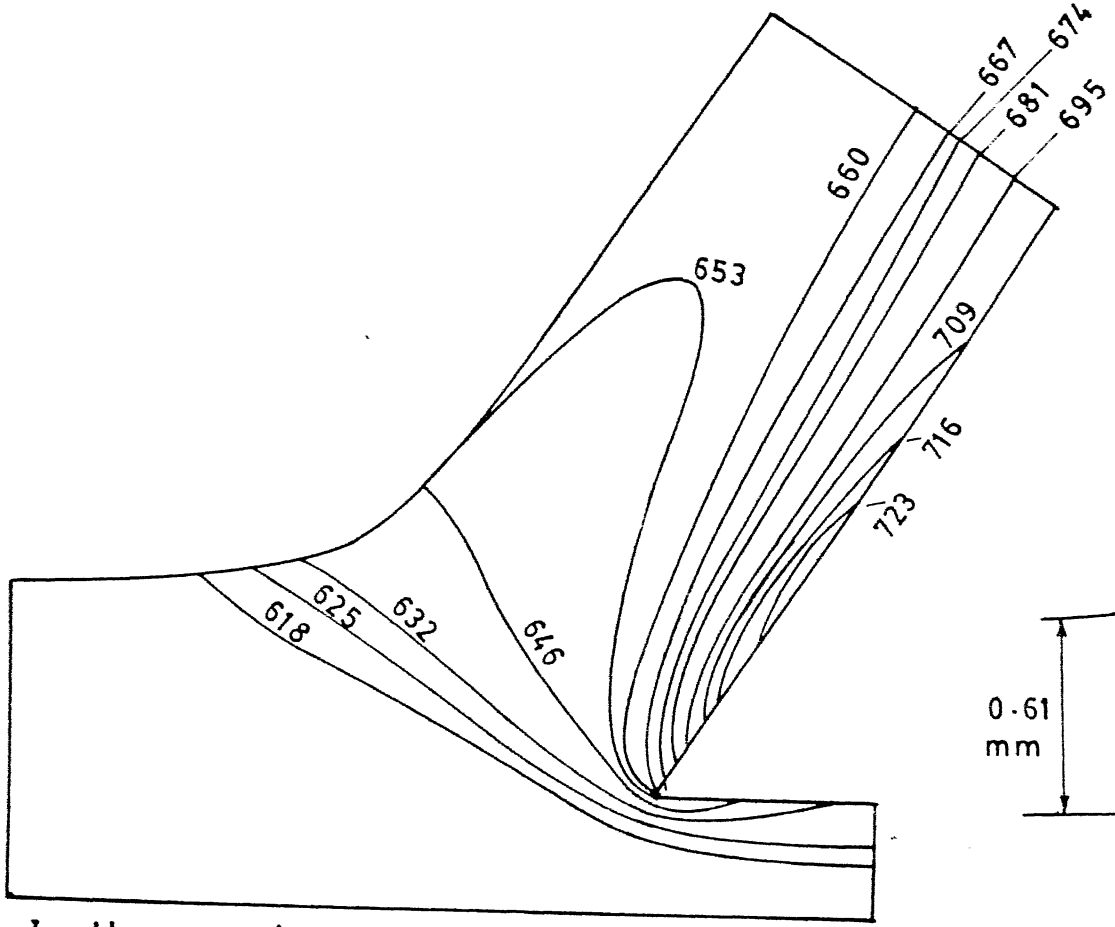
5.1 Isotherms and Deformation Patterns

The temperature values at all nodes have been predicted through the solution of the energy balance equation. Isotherms are then plotted by interpolating the available nodal temperatures, linearly on element boundaries, to obtain the locations where the given temperature values (isotherms) occur.

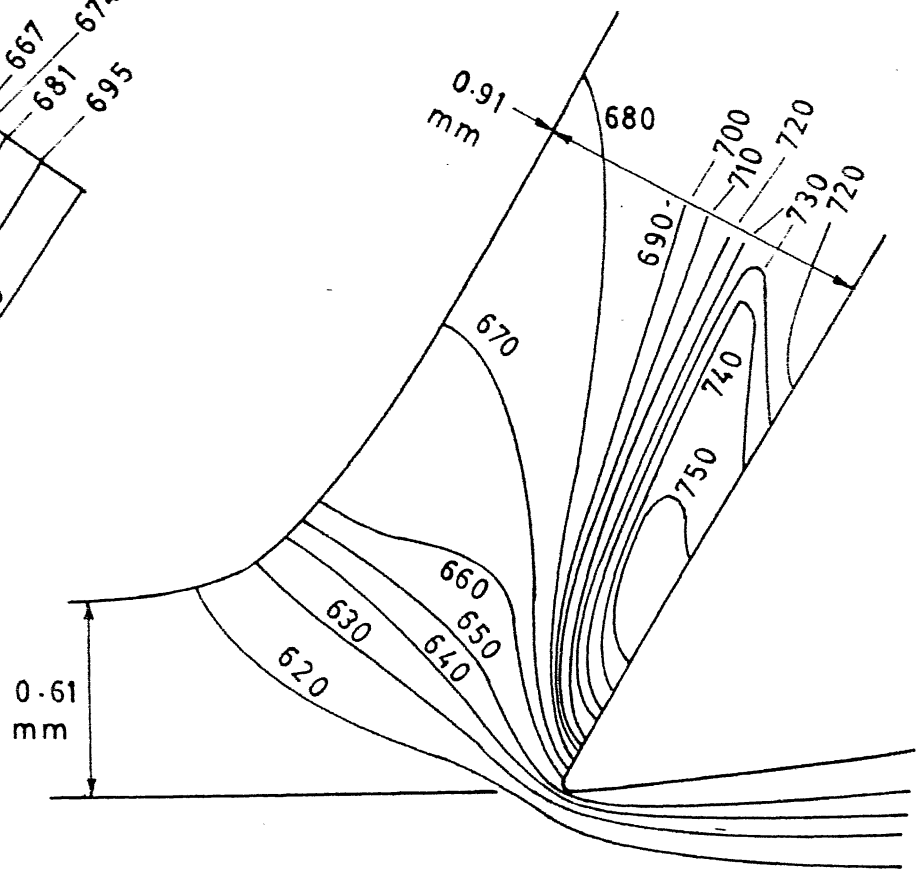
The strain rates ($\dot{\bar{\varepsilon}}$) at different points are first normalized, with respect to the maximum strain rate occurring in the solution domain. Points having the same value of normalized strain rates (after suitable interpolation) were connected by smooth curves and the curves of 3%, 4%, 5% and 8% normalized strain rates (as compared to the maximum of 100%) were thus plotted. These constant strain rate curves give an idea about the size and shape of the deformation zone.

Isotherms and deformation patterns for various cutting conditions are shown in Figures 5.1 to 5.5.

It is observed from the isothermal patterns that, as expected, the maximum temperatures occur near the chip-tool contact region. This has been observed by earlier researches as well. The reason for this is that, intense heating of a material by frictional work occurs at the contact region, in addition to earlier heating due to plastic work in PSDZ. It is also seen that temperatures in PSDZ are quite less than those occurring in SSDZ;



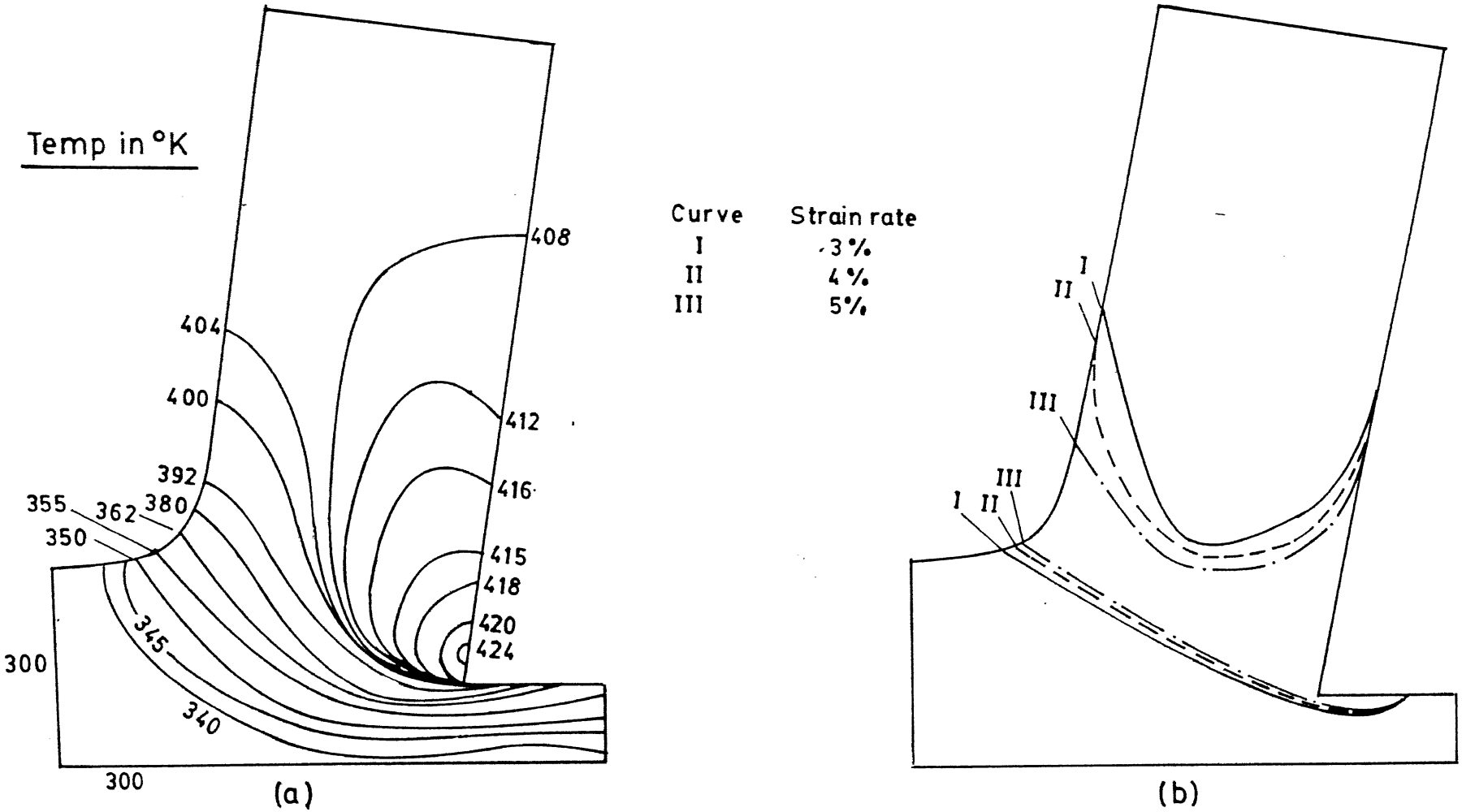
Isotherms obtained from present analysis



Isotherms observed by Boothroyd (19) using infrared photograph technique

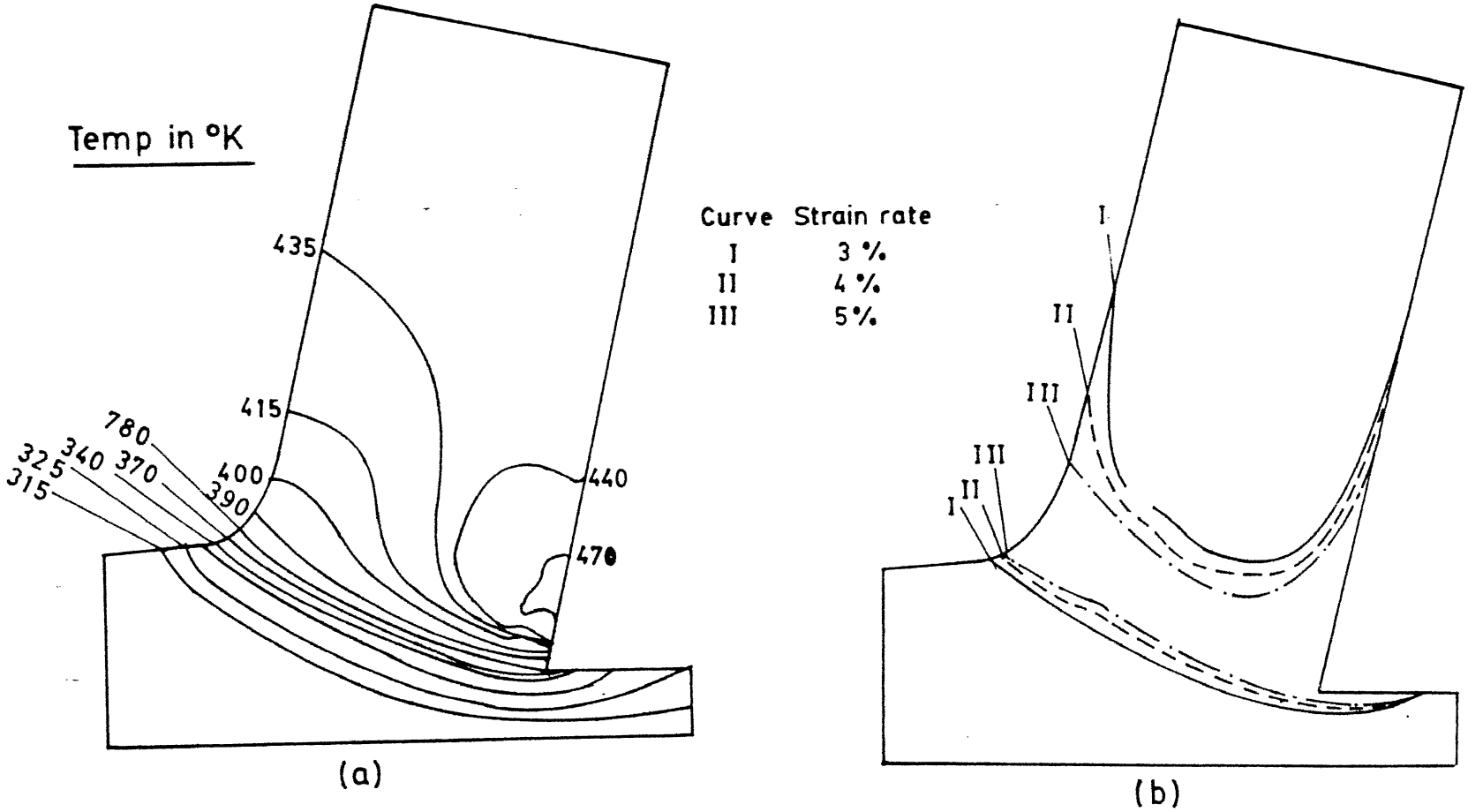
Workpiece matl. : Free cutting mild steel ; Cutting speed : 0.38 m/s (78 ft/min) ; Rake angle : 30° ; Width of cut : 6.35 mm ; Work piece temperature : 611°C

Fig. 5.0 Comparison with Boothroyd's (19) experimental results



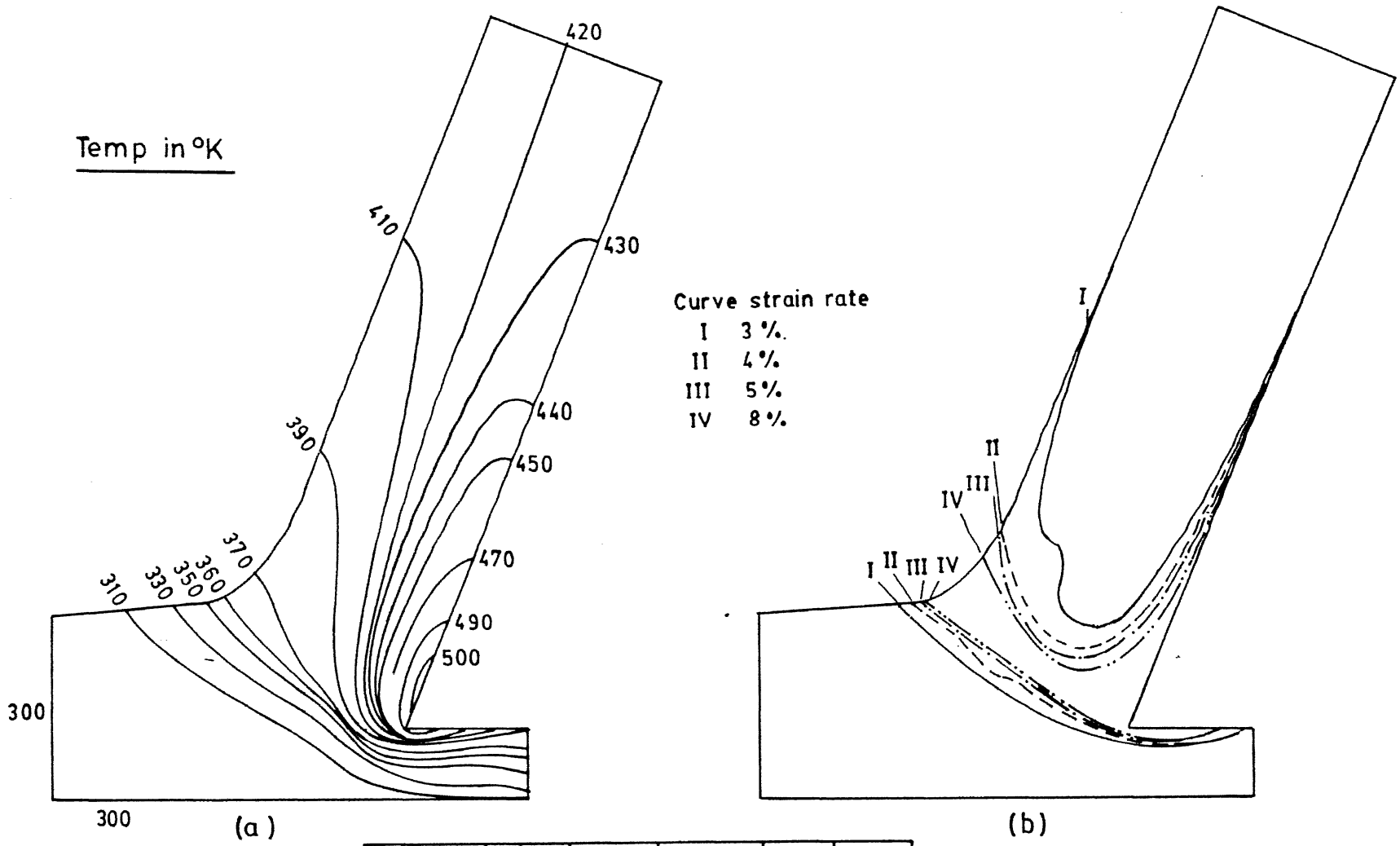
Test case	Rake	t (mm)	r = t/t_c	V (m/s)	m
2	10°	0·063	0·484	0·48	0·97

Fig. 5.2 (a) Isothermal pattern (b) Deformation pattern



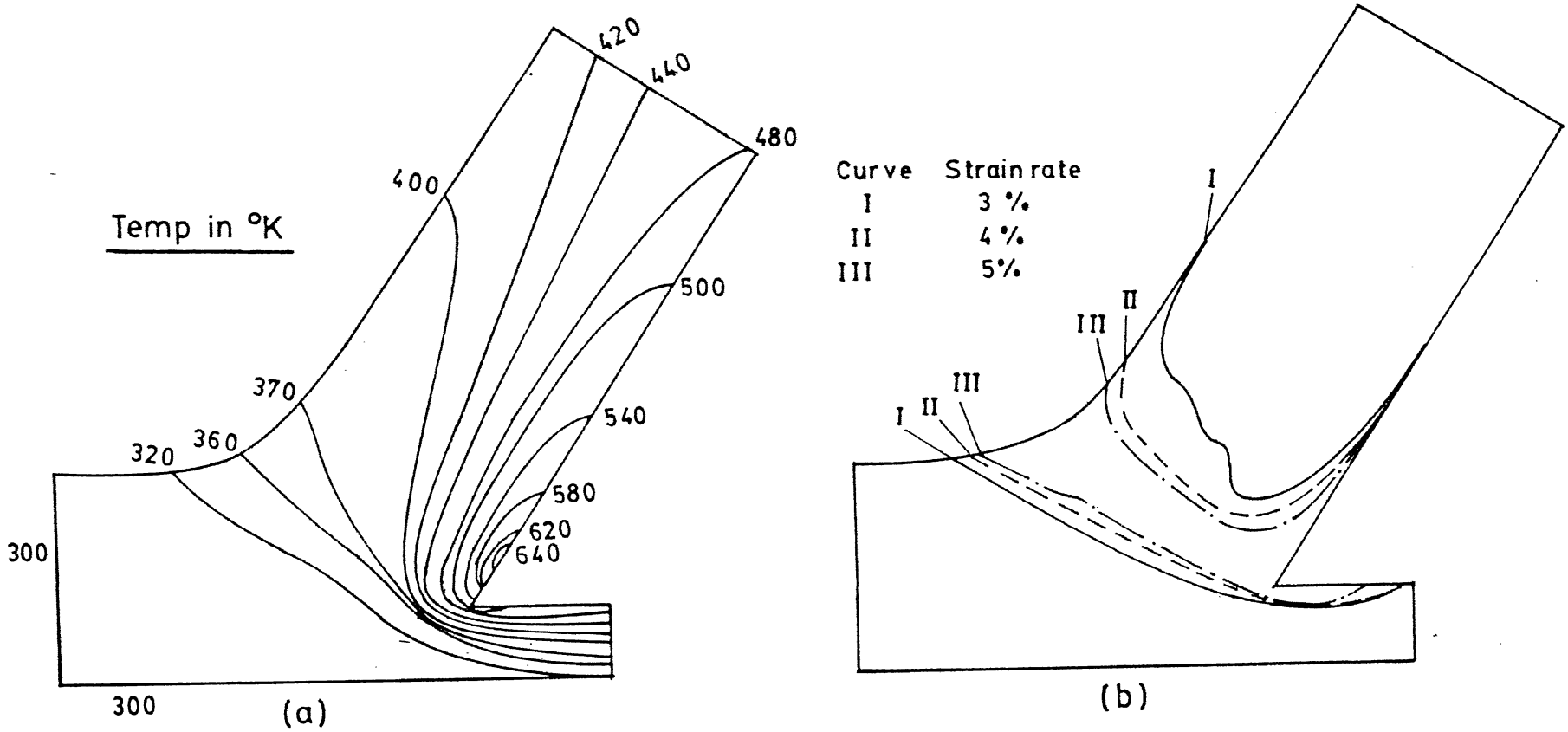
Test case	Rake	t (mm)	$r=t/t_c$	V (m/s)	m
4	10°	0·075	0·49	0·602	0·97

Fig. 5·3 (a) Isothermal pattern (b) Deformation pattern



Test case	Rake	t(mm)	$r = t/t_c$	V(m/s)	m
6	20°	0·3556	0·6634	0·411	0·97

Fig. 5·4 (a) Isothermal pattern (b) Deformation pattern



Test case	Rake	t (mm)	r = t/t_c	V (m/s)	m
7	30°	0·3556	0·619	0·413	0·97

Fig. 5.5 (a) Isothermal pattern (b) Deformation pattern

from this trend, it is clear that in metal cutting problems, frictional heat generation has an important effect on the temperatures. Bulk of the material, especially the domain nearer to uncut material is at room temperature. This can be attributed to the fact that a major portion of the heat produced is carried away by the chip. High temperatures are also observed in the freshly machined part of the workpiece, just below the tool tip. Some of the heat generated in PSDZ and SSDZ is conducted to the uncut material causing high temperature values just below the cutting edge; moreover, the relaxation processes which occur in that region also result in some heat generation. The occurrence of very high temperatures on the rake face contributes a lot to the tool wear.

Deformation patterns obtained in the present analysis confirm the well-established observations about the existence of a narrow PSDZ bounded by two parallel planes and a wedge-shaped SSDZ adjacent to the rake face. Among the various iso-strain rate curves, it appears that 5% cut-off gives a reasonable correspondence with the PSDZ thickness obtained through an established relation proposed by Jain & Gupta [27].

It is observed that the constant strain rate curves on the uncut material side are closely spaced compared to the cut material (i.e. chip). The reason for this may be that the material approaching the tool tip undergoes deformation near the shear plane, due to the shearing action caused by high compressive forces of the tool. For the chip the deformation occurs over a wider region since one of the surfaces is unrestrained. In a very small region of the uncut material i.e. just below the tool tip, a

close net of extended iso-strain rate lines is found. Since separation from the chip material occurs in this region, high stress values may be prevail which explain the above trend. The secondary deformation zone appears to be of triangular shape again confirming, a well established observation. Close-spacing of strain-rate lines in SSDZ is again attributed to high stresses in that region.

5.2 Effect of Process Variables

5.2.1 On tool Face temperatures :

It is observed from Figs. 5.6 to 5.8 that a non-uniform temperature distribution exists on the rake face and the maximum temperature always occurs at some distance away from the cutting edge. It can also be seen that, the temperature on the rake face rises sharply up to a point of maximum temperature with distance measured from the cutting edge and then drops moderately. Near the end of contact patch, the temperature more or less remains constant. This pattern has not been observed by earlier researchers. Muraka (17) observed that the temperature drops from a maximum near tool tip to a minimum at the point of chip separation. He had also observed a sharp change in the temperature gradient. Sarma (21) observed the occurrence of two local maxima. These differences may be attributed to the ignoring of effects such as thermal softening of material variations in thermo physical properties and friction factor variation along the contact patch, in the previous works.

An investigation into the effect of rake angle on rake face temperature reveals interesting trends. With increase in

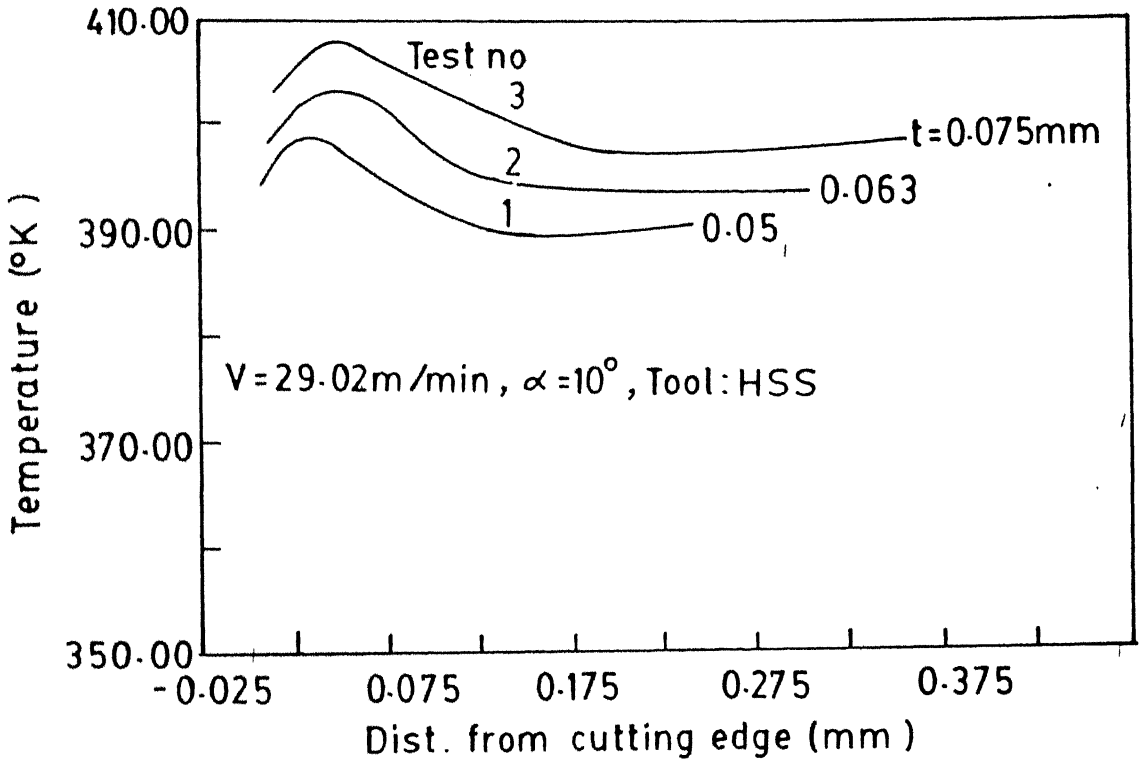


Fig. 5.6 Effect of depth of cut on rake face temperatures

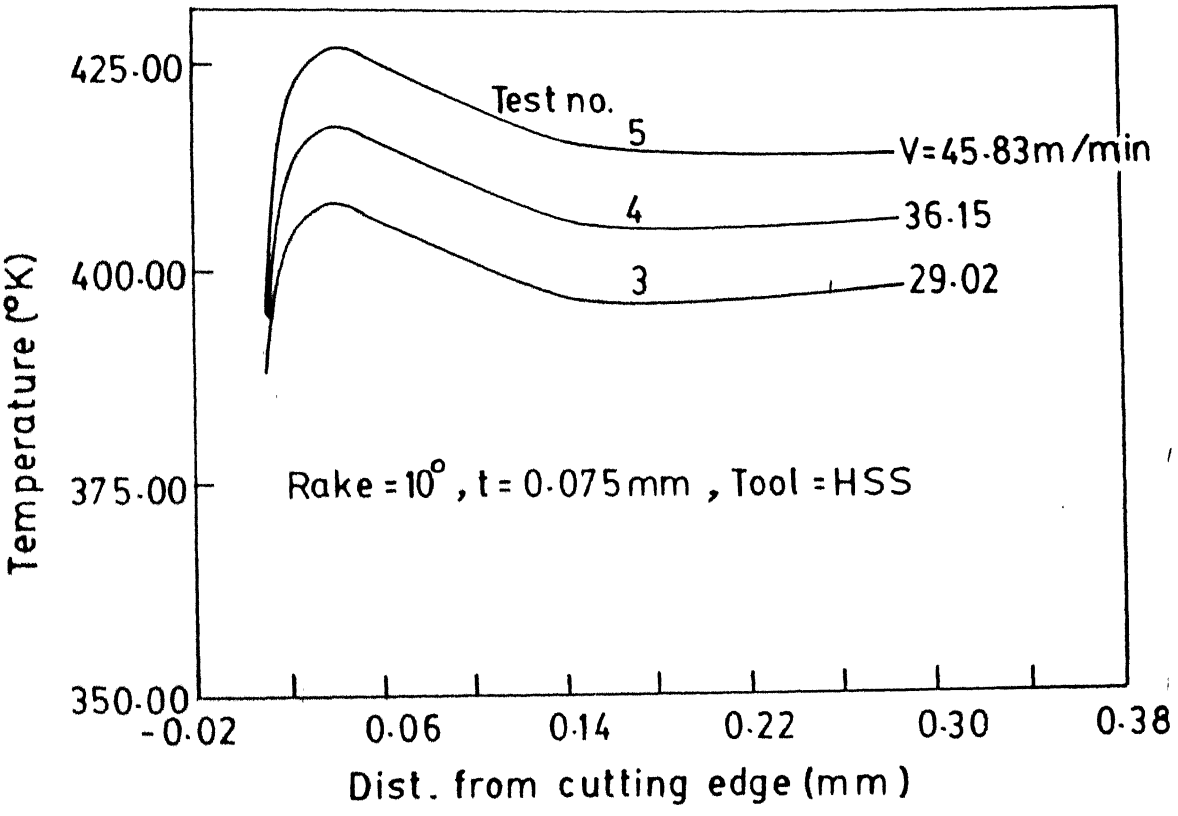


Fig. 5.7 Effect of cutting speed on rake face temperatures

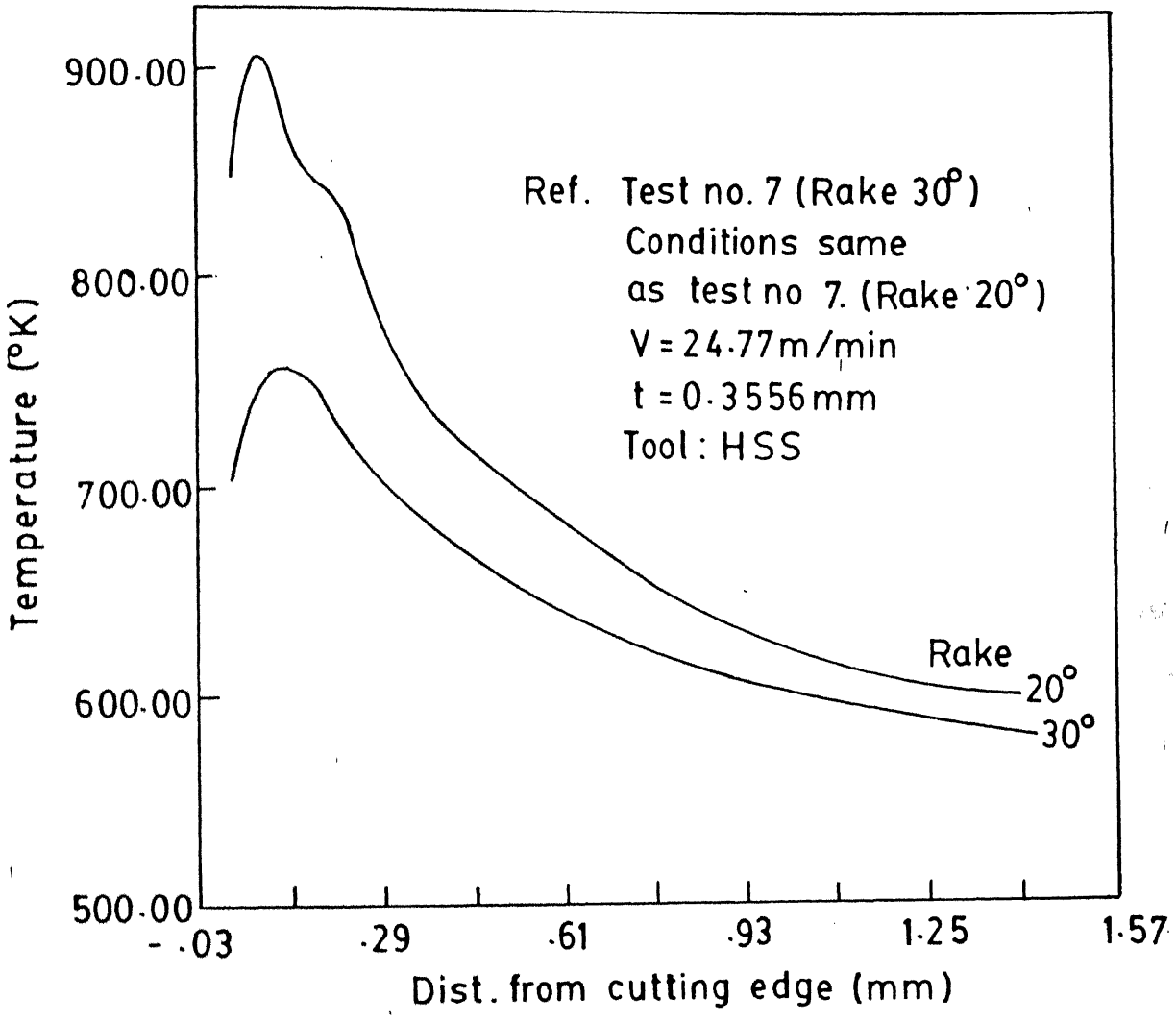


Fig. 5.8 Effect of rake angle on rake face temperatures

rake angle, the temperature distribution falls down. This is because, the included angle (i.e. angle between the tool flank and face) reduces with increase in rake angle. In turn, the total energy input to the system is reduced and hence the amount of heat generated is also decreased, thus causing a reduction in temperature.

Increasing the depth of cut or velocity causes an increase in the temperature distribution, proportionally. The reason for this is that increase in depth of cut or velocity leads to an increase in the energy input to the process.

5.2.2 On Temperature distribution in Shear Zone

Figures 5.9 to 5.12 show the effect of various process parameters on temperature distribution along and through the shear plane. The trends with respect to cutting parameters seems to be similar to those seen for the rake face temperature. The temperature values in the region near the tool tip are much higher than those near the free end. This can be attributed to the decrease in thermal conductivity of the material with increase in temperature, which restricts the outward flow of the heat generated in SSDZ for PSDZ. Secondly, the free surface being in contact with the atmosphere convective heat loss is expected. In the present work, shearplane is established as parallel to 10% cut-off strainrate curves, and in the mid of PSDZ, approximately.

5.2.3 Variation of Strain-rates in Shear Zone

Strain rate variation along and across the shear plane have been drawn in Figures (5.13 and 5.14). As can be expected, across the shear plane strain-rate rises rapidly, reaches a

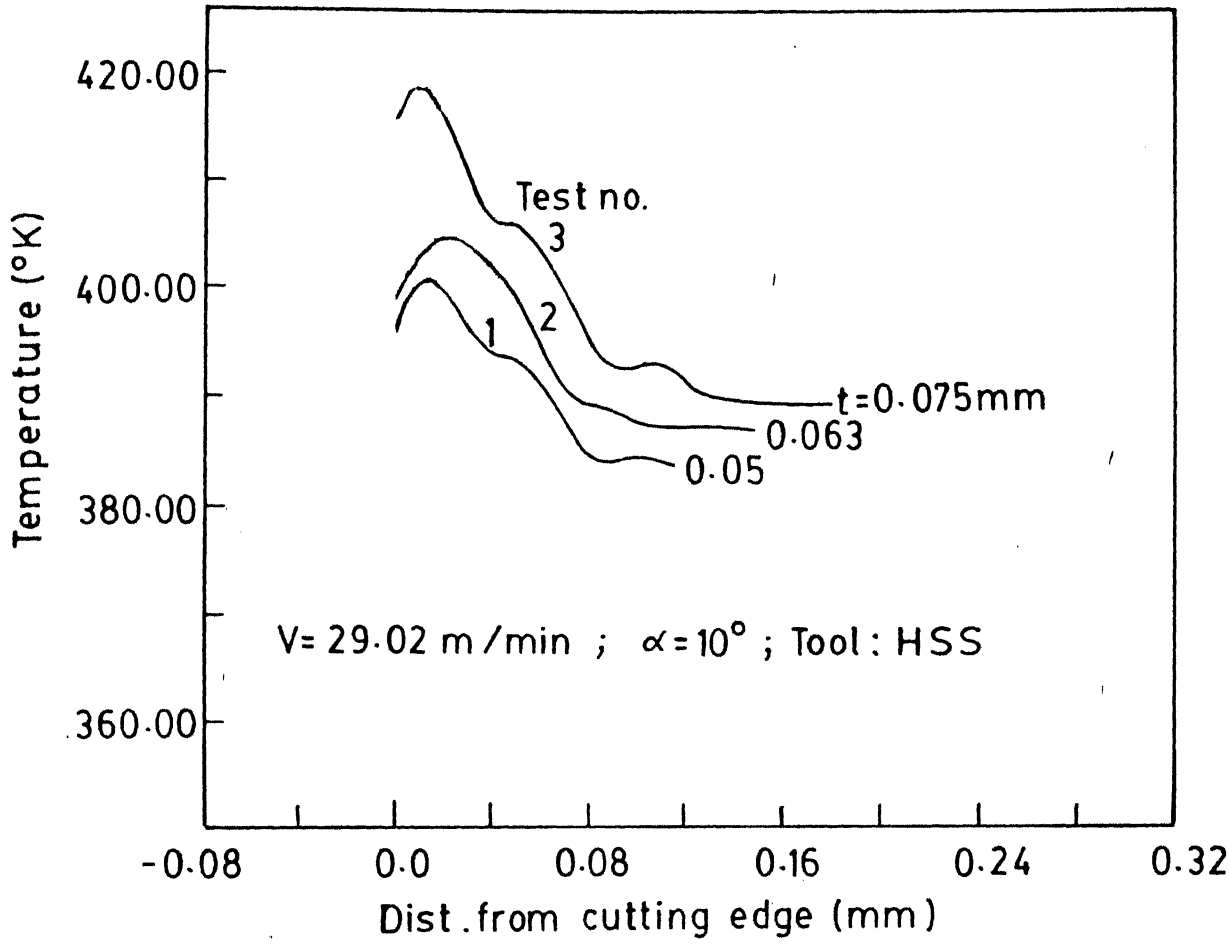


Fig. 5.9 Effect of depth of cut on temperature along shear plane

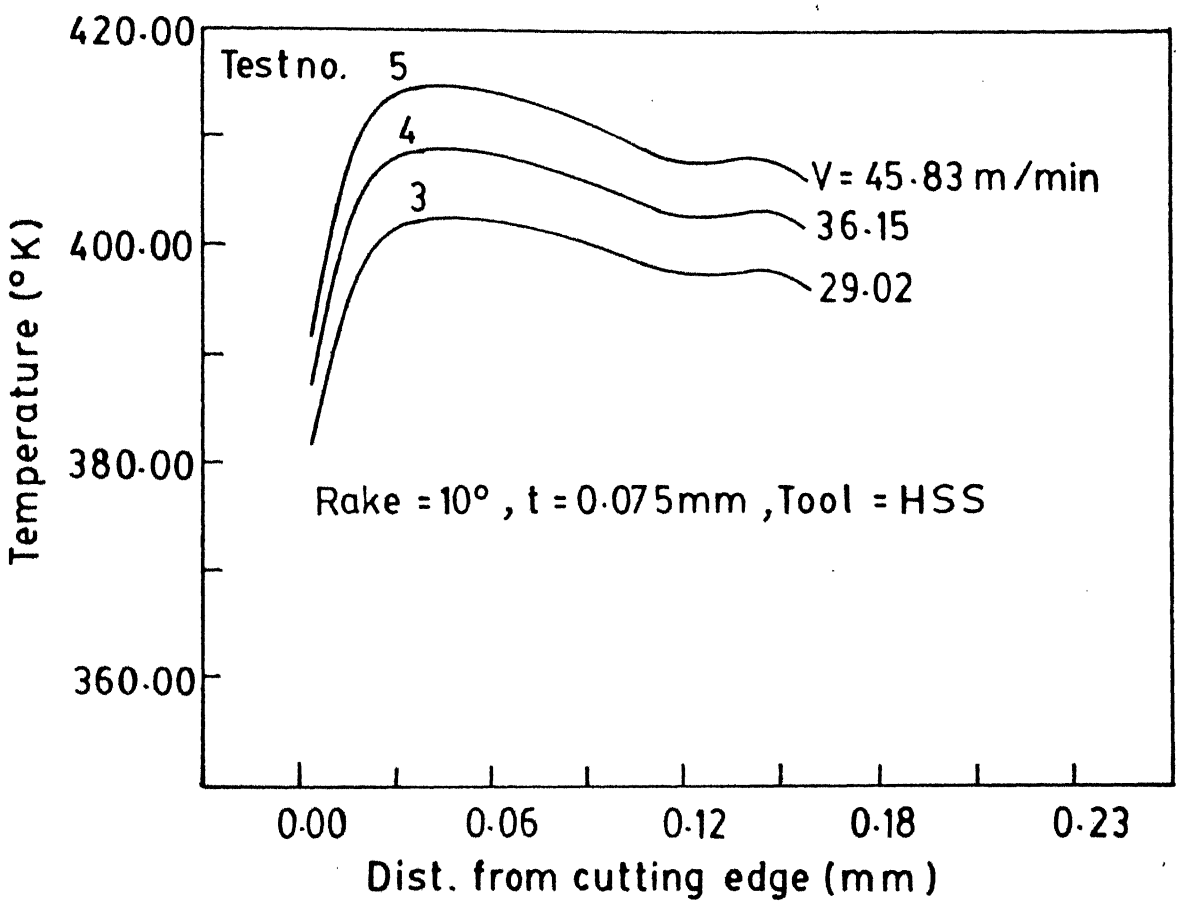


Fig. 5·10 Effect of cutting velocity on temperatures along shear plane

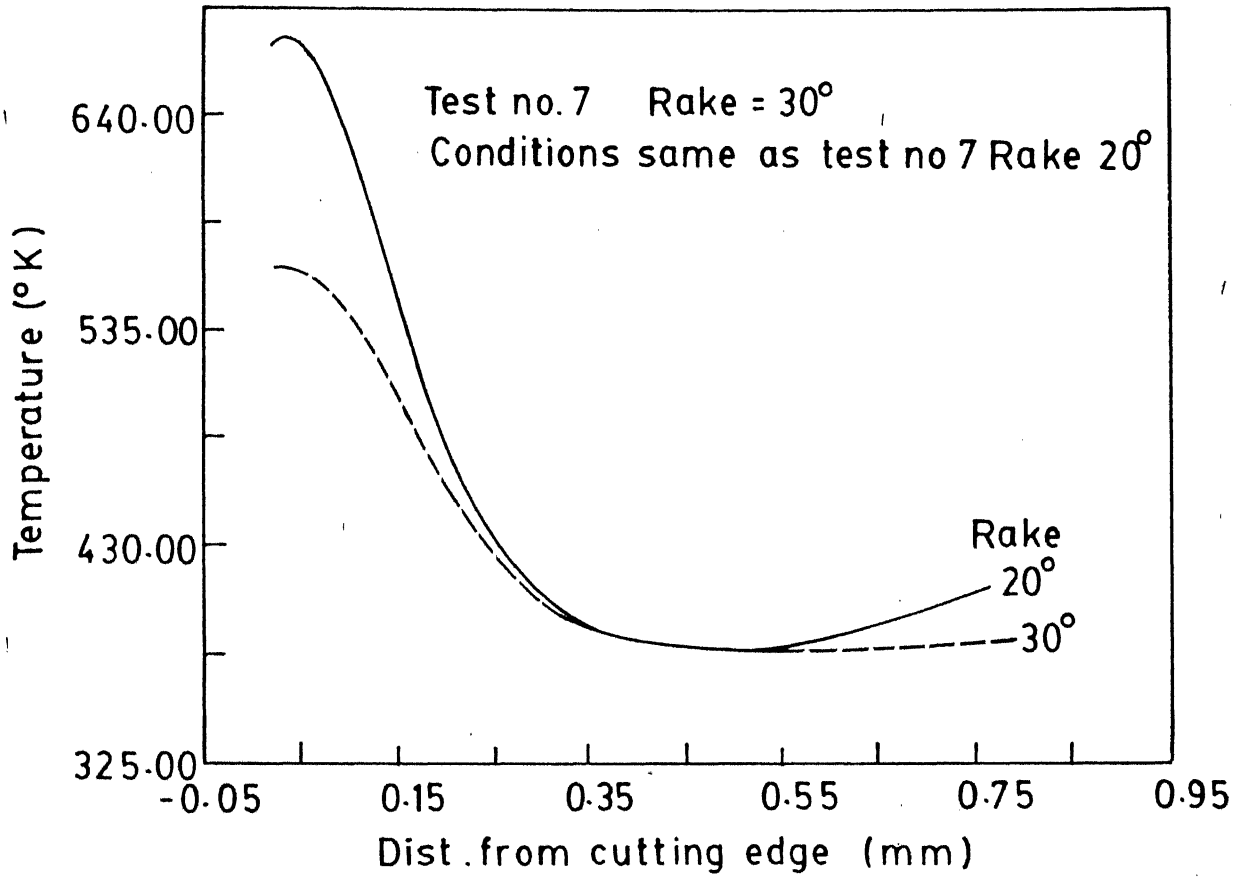


Fig. 5.11 Effect of rake angle on temperature distribution along shear plane

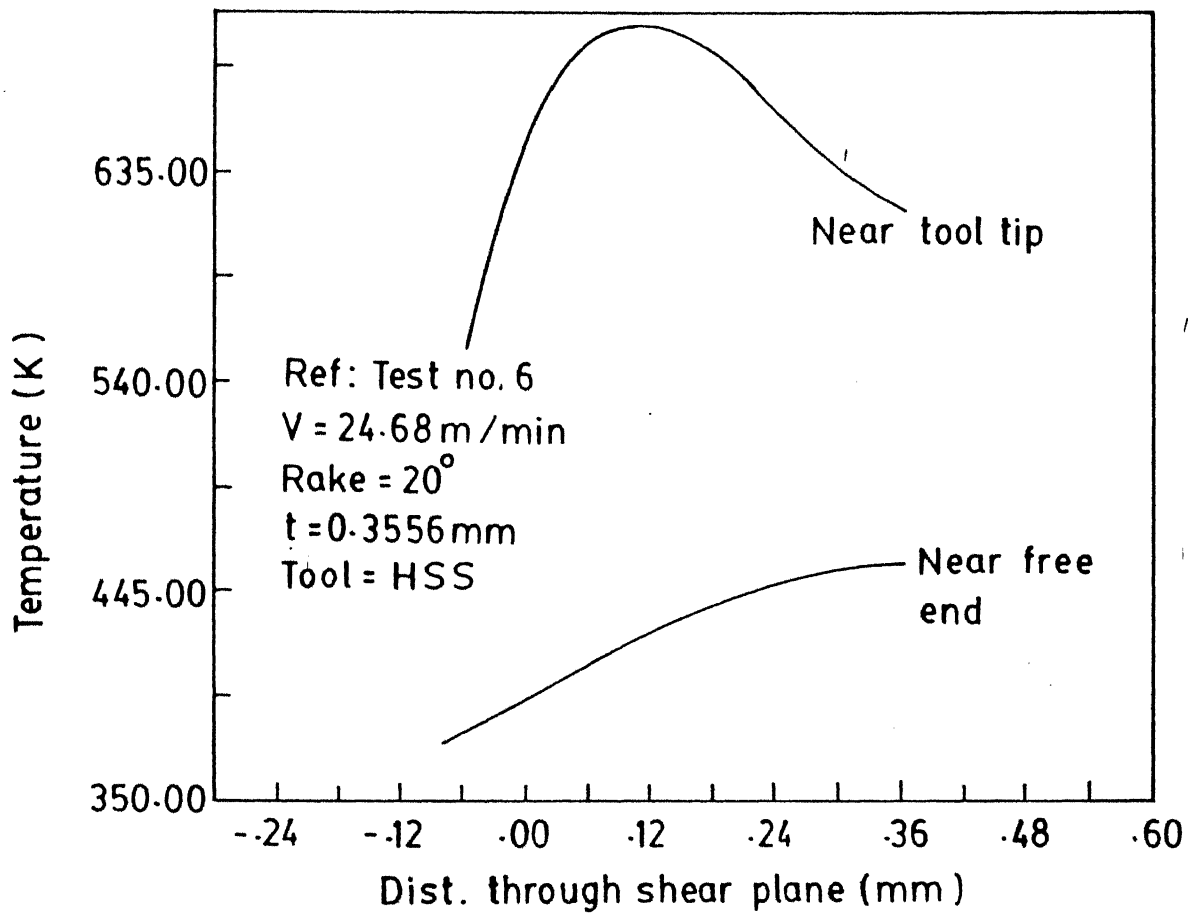


Fig. 5.12 Temperature distribution normal to the shear plane

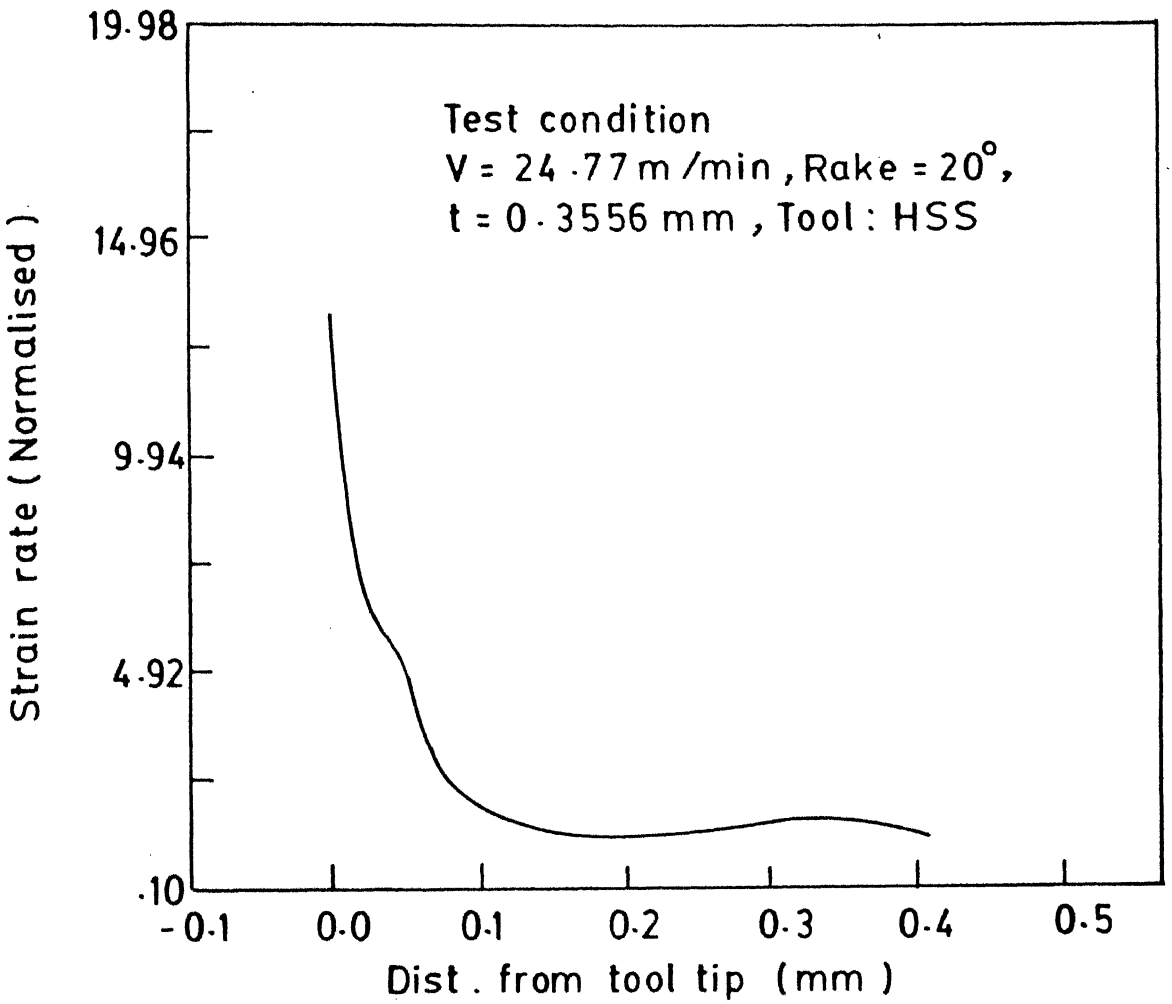


Fig. 5·13 Strain rate variation along shear plane

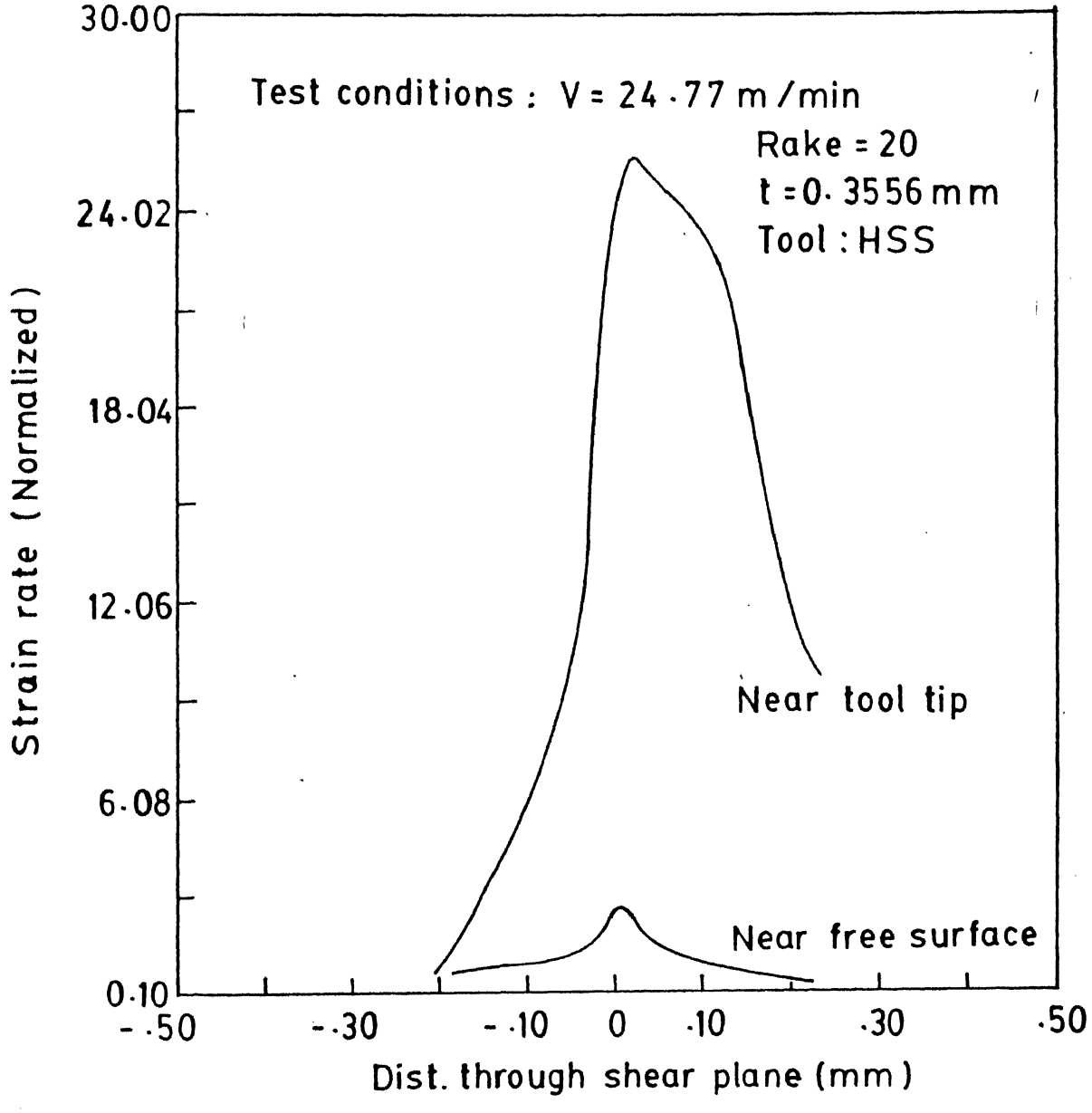


Fig. 5·14 Strain rate variation normal at shear plane

maximum somewhere near the shear plane and falls rapidly again. This confirms the existence of a narrow SSDZ near the shear plane. Also, it is observed that the strain rate is very high near the tool tip and less near the free surface or the middle part of the chip. Such a trend is to be anticipated, since SSDZ is also present near the tool tip.

Thus the results of present analysis have confirmed all the well-known trends in metal cutting to a reasonable extent. The width of PSDZ showed a slight decrease with the increase in cutting velocity, as observed in experimental studies.

5.3 Comparison with experimental Results

The primary objectives of the present analysis have been the prediction of cutting forces and the validation of the theoretical model using experimental results.

A comparison of the forces and temperatures obtained experimentally and theoretically is shown in Table. 2.

The forces (both cutting and feed forces) and temperatures have been predicted reasonably well. The analytical results deviate from experimental values just by 18-25%. Considering the complexities involved in the problem, the prescription of boundary conditions and the inherent, limitations of the visco-plastic model, the agreement seems to be quite satisfactory. The predicted temperature values are some what on the lower side when compared with the experimental values. This may be due to not having precise values of the friction factor on the chip-tool interface and heat transfer coefficient on the chip surface. It should also be kept in mind, however, that the tool

Table 2 : Comparison of analytical and experimental results

Test No.	Cutting Force (F_x) N		Feed Force (F_y) N		Average chip-tool interface temp. °K	
	Experi-mental	Analy-tical	Experi-mental	Analy-tical	Experi-mental	Analy-tical
1	563	461	389	313	469	398
2	641	582	416	346	481	412
3	702	562	452	351	502	445
4	667	535	453	349	500	456
5	633	514	449	334	502	458
6	3525	3618	1560	1820	-	463
7	2650	2258	545	869	-	561

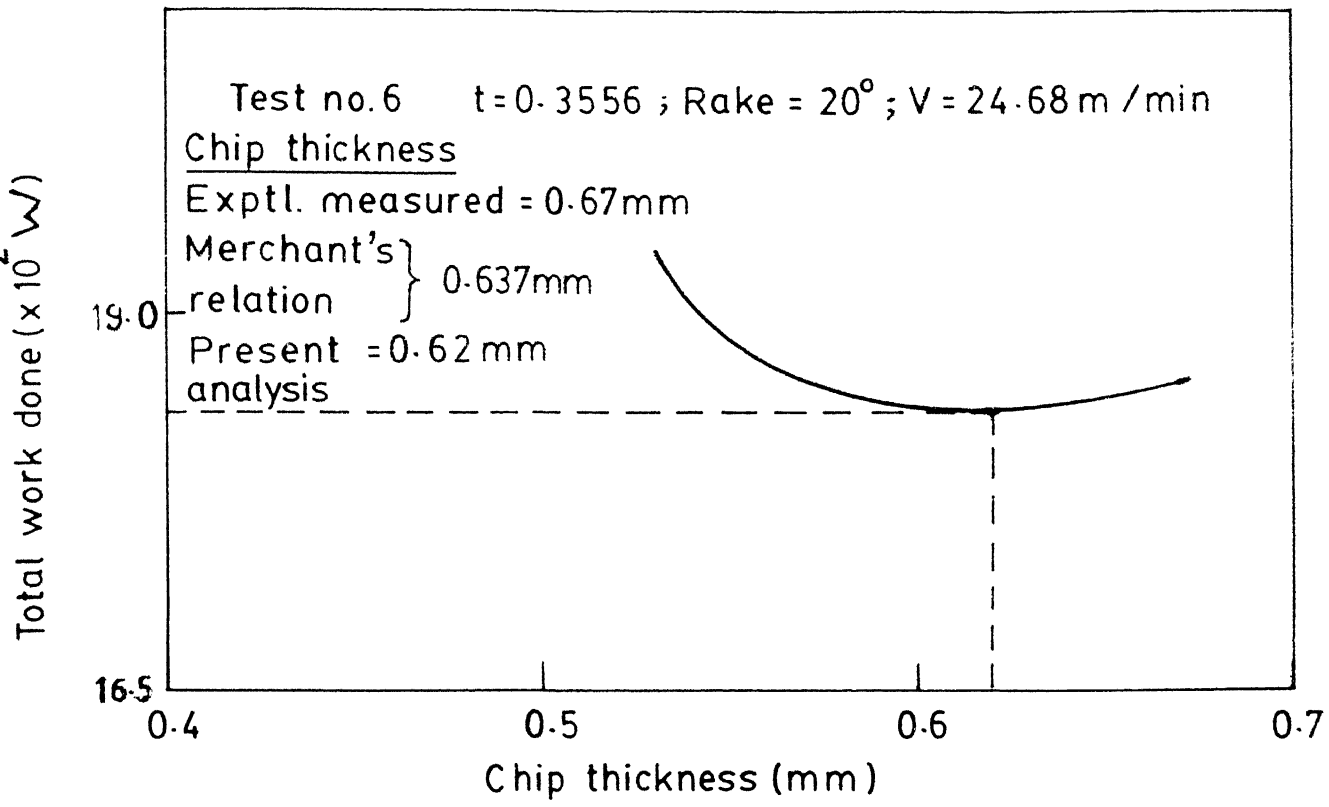


Fig. 5.15 Prediction of chip thickness from energy minimization principle

work thermocouple method used for experimentation does not give accurate results. Fig. 5.0 shows that the isotherms predicted reasonably match with the those obtained by Boothroyd (19) using infrared photographic technique. Slight deviation may be due to the above mentioned reasons.

5.4 Energy minimization Principle

The Classic Merchant's theory uses the energy minimization principle in establishing shear angle or the chip thickness ratio.

In the present analysis, the total work comprising of total plastic work and frictional work is obtained for various chip thicknesses. This was done generating separate grids for various chip thickness values, followed by the finite element analysis. It has been observed that the total work falls down with increase in chip thickness upto a certain value and then again rises. Thus, it can be stated that there exists a chip thickness value for which the total work or energy involved is a minimum. This chip thickness value matches reasonably with the predictions of the Merchant. The variation of total work with chip thickness is plotted in Fig. 5.15.

Thus, the present work provides an experimentally validated theoretical formulation and software package for the visco-plastic and thermal analysis of orthogonal metal cutting.

CHAPTER 6

CONCLUSIONS AND SUGGESTIONS

From the detailed finite element analysis of the orthogonal metal cutting, the following conclusions can be derived.

i) Viscoplastic model of the metal cutting can be successfully analysed using FEM to study the deformations and thermal phenomena thoroughly.

ii) Some significant quantities of practical importance such as forces, temperature distributions etc. can be predicted accurately. In fact, hardly any work has been done especially for estimating forces in metal cutting.

iii) The constant strainrate curves plotted with cut off values of 5%, 4%, 3% and 2% give a reasonably good idea about the sizes and shapes of the deformation zones. 5% cut off gives a better of PSDZ thickness.

iv) The effects of thermal softening and variations in material properties with the temperature are very important. After considering these variations only, the isotherm plots started showing good agreement with experimental measurements.

v) The parametric study conducted shows that the effects of depth of cut, cutting velocity and rake angles on temperature of strainrate distribution are significant.

vi) The thermal model developed confirms the energy minimization principle of metal cutting as proposed by Merchant for the

determination of the chip thickness ratio and shear angle.

Suggestions for Future Work

i) Similar analysis may be extended for a detailed stress analysis of the workpiece and tool.

ii) Tool wear phenomena can be analysed thoroughly using the temperature and force predictions.

iii) Effect of contact length on the deformation phenomena can be studied by using some of the new concepts such as controlled contact.

REFERENCE

1. Taylor, F.W., "On the Art of Cutting Metals", Trans. Amer. Soc. Mech. Engrs., Vol. 28, 1907, p. 31.

2. Merchant, M.E., "Mechanics of Metal Cutting Process", Pt. I & II, J. Appl. Phys., Vol. 16, 1945, p. 267.

3. Piispanen, V., "Theory of Formation of Metal chips", J. Appl. Phys., Vol. 19, 1948, p. 876.

4. Palmer, W.B. and Exley, P.L.B., "The Mechanics of orthogonal Machining", Proc. Instn. of Mech. Engrs, Vol. 173, 1954, p. 24.

5. Okushima, K. and Hitomi, K., "On cutting Mechanism of Soft Metals", Trans. Japan Soc. Mech. Engg., Vol. 23, No. 134, 1957, p. 674.

6. Kececioglu, D., "Shear Strain-rate in Metal Cutting and its effect on Shear flow Stress", Trans. Amer. Soc. Mech. Engrs., Vol. 80, 1958, p. 158.

7. Stevenson, M.G. and Oxley, P.L.B., "An Experimental Investigation on the Influence of the Speed and Scale on the Strainrate in a Zone of Intense Plastic Deformation", Proc. Instn. Mech. Engrs., Vol. 184, Pt. I, No. 31, 1969-70, p. 561.

8. Black, J.T., "Flow Stress Model in Metal Cutting", ASME, J. of Engg. for Industry, Vol. 95, No.4, 1973, p. 898.

9. Von-Turkovich, B.F., "Deformation Mechanics during adiabatic Shear", Proc. II North American Metal Working Res. Conference, Madison, Wisconsin, 1974, p. 862.

10. Von-Turkovich, B. and Micheletti, G.F., "Flow Zone Models in Metal Cutting", 9th Int. M.T.D.R. Conf., Univ. of Bermingham, 1968.

11. Zinkiewicz, O.C., Onate, E. and Heinrichs, "A General Formulation for Coupled Thermal Flow of Metals Using Finite Element", Int. J. Num. Meth. in Engg., Vol. 17, 1981, P. 1497.

12. Zienkiewicz, O.C., Jain, P.C. and Onate, E., "Flow of Solids during Forming the Extrusion, Some aspect of Numerical Solution", Int. J. Solids Struct., Vol. 14, 1978, p. 15.

13. Zienkiewicz, O.C. and Godbole, P.N., "Flow of Plastic and Visco-Plastic Solids with special reference to Extrusion and Forming Processes", Int. J. Num. Methods in Engg., Vol. 8....

14. Tay, A.O., Stevenson, M.G. and de Vahl Davis, G., "Using the Finite Element Method to determine temperature Distributions in Orthogonal Machining", Proc. Inst. of Mech. Engrs, Vol. 188, 1974, p. 627.

15. Tay, A.O., Stevenson, M.G. deVahl Davis, G., and Oxley, P.L.B., "A Numerical Method for Calculating Temperature Distribution in Machining From Force and Shear Angle Measurements", Int. J. of MTDR, Vol. 16, 1976, p. 335.

16. Stevenson, M.G., Wright, P.K. and Chow, J.S., "Furthur Developments in Applying the finite Element Method to the Calculation of Temperature Distributions in Machining and Comparisons with Experiment", Trans. ASME, J. of Engg. for Industry, Aug. '83, Vol. 105, p. 149.

17. Muraka, P.D., Borrow, G., and Hindiya, S., "Influence of the process variables on the temperature Distribution in Orthogonal Machining Using Finite Element Method", Int. J. of Mech. Sci., Vol. 21, 1979, p. 445.

18. H.S. Balaji, "An Application of FEM to the Estimation of Temperature Field in Machining with Coated Carbide Tools", M.Tech. Thesis, Dept. of Mech. Engg., IIT, Kanpur. 1986

19. Boothroyd, G., "Fundamentals of Metal Machining and Machine Tools", 1975, Scripta Book Company Washington, D.C. (McGraw-Hill).

20. Venkatesh, Y.C. and Chandrasekaran, H., "Experimental Techniques in Metal cutting", 1987, Prentice-Hall of India. Pvt. Ltd., New Delhi.

21. C. Mullikarjuna Sarma, "A Finite Element Analysis of Thermal and Deformation Processes in Metal Cutting", M.Tech. Thesis, Dept. of Mech. Engg., IIT, Kanpur 1990.

22. Juneja, B.L. and Sekhon, G.S., "Fundamentals of Metal Cutting and Machine Tools". 1987, Wiley Eastern Ltd., India.

23. Armarego, E.J.A. and Brown, R.H., "The Machining of Metals", 1969, Prentice-Hall, Inc. USA.

24. Taylor, C. and Hughes, T.G., "Finite Element Programming of Navier Stokes Equations", Ist Ed., 1981, Pine ridge Press Ltd., U.K.

25. Kobayashi, S., Soo-Ikoh and Taylan Altan, "Metal Forming and the Finite Element Method", 1989, Oxford University Press.

26. Rao, S.S., "The Finite Element Method in Engineering", IInd Ed., 1989, Peragamon Press.

27. Jain, V.K. and Gupta, B.K. "Effects of Accelerated Tests on Shear Flow Stress in Machining", Trans. ASME, J. of Engg. For Industry, Aug 1987, Vol. 109, p. 206.

28. Metals Handbook (ASME) p. 627.

29. Hahn, R.S., Discussion of Paper by M.C. Shaw, N.H. Cook and Io Fini "Shear Angle Relationship in Metal Cutting", Trans. ASME, series B, Vol. 83, p. 181.

30. I. Imafuku and K. Chijiwa, Mitsui Engg. and Ship Building Co. Ltd., Okayama, Japan, "A Methematical Model for Shrinkage Cavity Prediction in steel castings", AFS Transactions, V 86, p. 527, 1987.

31. Komvopowos, K. and Erpenbeck, S.A., "Finite Element Modelling of Orthogonal Metal Cutting", Trans. ASME, J. Engg. for Industry, Aug. 91, V. 113. P. 213.

```
C ##################################################################################
C ###################PROGRAM FOR GRID GENERATION####################################
C ##################################################################################
C
      DIMENSION XNOD1(106),YNOD1(106),XNOD2(293),YNOD2(293)
      DIMENSION COORD(386,2)
      COMMON/A1/LNODS(111,8),LNODS1(27,8),LNODS2(84,8)
      COMMON/A2/BOUDC1(48,2),BOUDC2(80,2)
      COMMON/A5/RAKEAN,DCUT,CHPTHK,CONLEN,COSA,SINA
      COMMON/A11/NX1,NY1,NELEM1,NPOIN1
      COMMON/A12/NX2,NY2,NELEM2,NPOIN2
      COMMON/A13/NELEM,NPOIN,NELCOM,NNODP
C
      OPEN(UNIT=28,FILE='AUTOG.OUT')
      OPEN(UNIT=29,FILE='COORD.OUT')
      OPEN(UNIT=30,FILE='CONNECT.OUT')
      OPEN(UNIT=24,FILE='SUPERG.OUT')
C
      PI=3.1415927
        WRITE(*,1)
    1 FORMAT(/' ENTER DEPTH OF CUT')
        READ(*,*) DCUT
        WRITE(*,2)
    2 FORMAT(/' ENTER RAKE ANGLE')
        READ(*,*) RAKEAN
      RAKE=RAKEAN*PI/180.0
      COSA=COS(RAKE)
      SINA=SIN(RAKE)
        WRITE(*,7)
    7 FORMAT(/' FOR CHIP THK & CONTACT LENGTH',
     !       //' TYPE 1 IF YOU ARE GIVING DATA.',
     !        /' TYPE 0 FOR CALCULATING BY ITSELF',
     !       //'   NOW ENTER YOUR OPTION---')
        READ(*,*)OPT
        IF(OPT.EQ.1)GOTO 444
        WRITE(*,8)
    8 FORMAT(/' ENTER FRICTION FACTOR ')
        READ(*,*)FRIFAC
        BETA=ATAN(FRIFAC)
        FI=3.142/4+0.5*(RAKE-BETA)
        CHR=TAN(FI)/(COSA+SINA*TAN(FI))
        CHPTHK=DCUT/CHR
        CONLEN=DCUT*SIN(FI+BETA-RAKE)/(SIN(FI)*COS(BETA))
        WRITE(*,9) CHR,CHPTHK,CONLEN
    9 FORMAT(//' CALCULATED VALUES ARE'
     !       //' CHIP THK RATIO = ',F8.4,
     !        /' CHIP THK       = ',F8.4,
     !        /' CONTACT LENGTH = ',F8.4)
        PAUSE'PRESS RETURN TO EXECUTE'
        GOTO 555
  444   WRITE(*,3)
    3 FORMAT(/' ENTER CHIP THICKNESS')
        READ(*,*) CHPTHK
        WRITE(*,4)
    4 FORMAT(/' ENTER CONTACT LENGTH ')
```

```
      READ(*,*)CONLEN
  555 CHPTHK=CHPTHK/DCUT
      CONLEN=CONLEN/DCUT
      NX1=9
      NY1=3
      NX2=14
      NY2=6
      NNODP=8
      NELCOM=6
      NODCOM=2*NELCOM+1
      NELEM1=NX1*NY1
      NELEM2=NX2*NY2
      NELEM=NELEM1+NELEM2
      NPOIN1=(2*NX1+1)*(2*NY1+1)-NX1*NY1
      NPOIN2=(2*NX2+1)*(2*NY2+1)-NX2*NY2
      NPOIN=NPOIN1+NPOIN2-(NELCOM*2+1)
      DO 300 I=1,NPOIN
      COORD(I,1)=0.0
      COORD(I,2)=0.0
  300 CONTINUE
      ICALL=1
      NTOTC1=2*NPOIN1
      NBCON1=(NX1+NY1)*4
      CALL AUTO(NX1,NY1,NBCON1,XNOD1,YNOD1,NTOTC1,ICALL)
      ICALL=ICALL+1
      NTOTC2=2*NPOIN2
      NBCON2=(NX2+NY2)*4
      CALL AUTO(NX2,NY2,NBCON2,XNOD2,YNOD2,NTOTC2,ICALL)
      CALL SUPER(LNODS,NSIF1)
      DO 10 I=1,NPOIN1
      COORD(I,1)=XNOD1(I)
      COORD(I,2)=YNOD1(I)
   10 CONTINUE
      JJ=0
      DO 20 I=NODCOM+1,NPOIN2
      JJ=JJ+1
      COORD(NPOIN1+JJ,1)=XNOD2(I)
      COORD(NPOIN1+JJ,2)=YNOD2(I)
      NN=NPOIN1+JJ
   20 CONTINUE
      WRITE(29,56)
   56 FORMAT(//,6X,'NODE',3X,'X-COORD',4X,'Y-COORD',/)
      DO 1289 I=1,NPOIN
        COORD(I,1)=COORD(I,1)*DCUT
        COORD(I,2)=COORD(I,2)*DCUT
      WRITE(29,108)I,COORD(I,1),COORD(I,2)
  108 FORMAT(6X,I5,2F10.4)
 1289 CONTINUE
        TYPE*,'PL.SEE COORD.OUT FOR COORDINATES:'
      STOP
      END
C---1---------------------------------------------------------------
C
      SUBROUTINE SUPER(LNODS,NSIF1)
C
```

```
      DIMENSION LNODS(111,8),LNODS1(27,8),LNODS2(84,8),LNODSX(111,8)
      COMMON/A11/NX1,NY1,NELEM1,NPOIN1
      COMMON/A12/NX2,NY2,NELEM2,NPOIN2
C
      DO 600 I=1,NELEM1
      DO 600 J=1,8
  600 LNODS1(I,J)=0
      DO 610 I=1,NELEM2
      DO 610 J=1,8
  610 LNODS2(I,J)=0
      DO 630 I=1,NELEM
      DO 630 J=1,8
  630 LNODS(I,J)=0
      WRITE(24,26) NELCOM
   26 FORMAT(/10X,'NO.OF COMMON ELEMENTS =',I4,/)
        NPOINX=NPOIN1
        NELEMX=NELEM1
      CALL ZONE(NX1,NY1,NPOINX,NELEMX,LNODSX)
        DO 99 KK=1,NELEMX
        DO 88 JJ=1,8
        LNODS1(KK,JJ)=LNODSX(KK,JJ)
   88 CONTINUE
   99 CONTINUE
      WRITE(24,22) NX1,NY1,NELEM1,NPOIN1
   22 FORMAT(//5X,'NX1=',I3,2X,'NY1=',I3,2X,'NELEM1=',I5,
     !2X,'NPOIN1=',I5,/)
      WRITE(24,33)
   33 FORMAT(//10X,'ELEMENT CONNECTIVITY INFO. FOR ZONE1',
     !//,6X,'IELEM',16X,'LOCAL NODE NOS',/,
     !16X,'1',5X,'2',5X,'3',5X,'4',5X,'5',4X,'6',6X,'7',5X,'8',/)
      DO 500 IELEM=1,NELEM1
      WRITE(24,11) IELEM,(LNODS1(IELEM,JJ),JJ=1,8)
   11 FORMAT(5X,I5,2X,8(X,I5))
  500 CONTINUE
        NPOINX=NPOIN2
        NELEMX=NELEM2
C
      CALL ZONE(NX2,NY2,NPOINX,NELEMX,LNODSX)
C
        DO 999 KKK=1,NELEMX
        DO 888 JJJ=1,8
        LNODS2(KKK,JJJ)=LNODSX(KKK,JJJ)
  888 CONTINUE
  999 CONTINUE
      WRITE(24,23) NX2,NY2,NELEM2,NPOIN2
   23 FORMAT(//5X,'NX2=',I3,2X,'NY2=',I3,2X,'NELEM2=',I5,
     !2X,'NPOIN2=',I5/)
      WRITE(24,34)
   34 FORMAT(//10X,'ELEMENT CONNECTIVITY INFO. FOR ZONE2',
     !//,6X,'IELEM',16X,'LOCAL NODE NOS',/,
     !16X,'1',5X,'2',5X,'3',5X,'4',5X,'5',4X,'6',6X,'7',5X,'8',/)
      DO 510 IELEM=1,NELEM2
      WRITE(24,11) IELEM,(LNODS2(IELEM,JJ),JJ=1,8)
  510 CONTINUE
      DO 100 IELEM=1,NELEM1
```

```
      DO 150 INODP=1,NNODP
      LNODS(IELEM,INODP)=LNODS1(IELEM,INODP)
  150 CONTINUE
  100 CONTINUE
      NSIF1=NPOIN1-(2*NX1+1)
      NSIF2=NSIF1+2*(NX1-NELCOM)
      DO 220 IELEM=1,NELEM2
      DO 250 INODP=1,NNODP
      NSHIFT=NSIF2
      IF((IELEM.LE.NELCOM).AND.(INODP.LE.3))NSHIFT=NSIF1
      LNODS(IELEM+NELEM1,INODP)=LNODS2(IELEM,INODP)+NSHIFT
      I1=IELEM+NELEM1
  250 CONTINUE
  220 CONTINUE
      WRITE(24,25) NELEM,NPOIN
   25 FORMAT(//5X,'NELEM=',I5,2X,'NPOIN=',I5/)
      WRITE(24,35)
   35 FORMAT(//10X,'ELEMENT CONNECTIVITY INFO. FOR DOMAIN',
     !//,6X,'IELEM',16X,'LOCAL NODE NOS',/,
     !16X,'1',6X,'2',6X,'3',6X,'4',6X,'5',5X,'6',7X,'7',6X,'8',/)
      DO 520 IELEM=1,NELEM
      WRITE(24,11) IELEM,(LNODS(IELEM,JJ),JJ=1,8)
  520 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
C
      SUBROUTINE ZONE(NX,NY,NPOINX,NELEMX,LNODSX)
      DIMENSION LNODSX(111,8)
      DO 10 I=1,NELEMX
      DO10 J=1,8
   10 LNODSX(I,J)=0
      DO 100 IELEM=1,NELEMX
      IY=IELEM/NX
      Y=FLOAT(IELEM)/NX
      D=Y-IY
      IF (D.EQ.0.0) IY=IY-1
      IX=IELEM-IY*NX
      LNODSX(IELEM,1)=IY*(3*NX+2)+(IX-1)*2+1
      LNODSX(IELEM,2)=LNODSX(IELEM,1)+1
      LNODSX(IELEM,3)=LNODSX(IELEM,2)+1
      LNODSX(IELEM,4)=LNODSX(IELEM,3)+2*NX-IX+1
      LNODSX(IELEM,5)=LNODSX(IELEM,4)+NX+IX+1
      LNODSX(IELEM,6)=LNODSX(IELEM,5)-1
      LNODSX(IELEM,7)=LNODSX(IELEM,6)-1
      LNODSX(IELEM,8)=LNODSX(IELEM,4)-1
  100 CONTINUE
      RETURN
      END)
C-----------------------------------------------------------------------
C
C
      SUBROUTINE AUTO(NXEL,NYEL,NBCON,XNODX,YNODX,NTOTXY,ICALL)
      DIMENSION TRSPOL(2000)
      INTEGER BNOD(1000)
```

```
      DIMENSION XNODX(2000),YNODX(2000)
      COMMON/AREA2/NX,NY,NOD,NEL,NDXDIR,NDYDIR
      NX=NXEL
      NY=NYEL
      NDXDIR=2*NX+1
      NDYDIR=2*NY+1
      NBCON=4*(NX+NY)
      NEL=NX*NY
      NOD=(2*NX+1)*(2*NY+1)-NEL
      CALL INPUT(NBCON,NOD,ICALL)
      CALL BDYNOD(NBCON,BNOD)
      CALL INTPOL(NDXDIR,NDYDIR,NBCON,NTOTXY,BNOD,TRSPOL,ICALL)
      DO 133 I=1,NBCON
  133 CONTINUE
      LAKE=1
      DO 53 IBM=1,NOD
      XNODX(IBM)=TRSPOL(LAKE)
      YNODX(IBM)=TRSPOL(LAKE+1)
      LAKE=LAKE+2
   53 CONTINUE
      DO 1289 I=1,NOD
 1289 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------------
C
      SUBROUTINE INPUT(NBCONX,NODX,ICALL)
      INTEGER BNODX(1000)
      COMMON/A2/BOUDC1(48,2),BOUDC2(80,2)
      COMMON/AREA2/NX,NY,NOD,NEL,NDXDIR,NDYDIR
      COMMON/A4/X0,Y0,X1,Y1,X2,Y2
      COMMON/A5/RAKEAN,DCUT,CHPTHK,CONLEN,COSA,SINA
      NNODP=8
      I3=7
      IF(ICALL.EQ.1) GOTO 500
      IF(ICALL.EQ.2) GOTO 600
  500 CONTINUE
      I1=0
      I2=29
      DX1=2./6.
      DX2=DX1
C\FIRST BOUND COORD.& SECOND
      XCOR1=-3.0
      YCOR1=-0.6
      XCOR2=XCOR1
      YCOR2=0.0
      DO 10 I=1,7
      I1=I1+1
      I2=I2+1
      BOUDC1(I1,1)=XCOR1+FLOAT(I-1)*DX1
      BOUDC1(I1,2)=YCOR1
      BOUDC1(I2,1)=XCOR2+FLOAT(I-1)*DX2
      BOUDC1(I2,2)=YCOR2
   10 CONTINUE
      I4=36
```

```
      DX3=2.0/12
      DX4=DX3
      XCOR3=-1.0
      YCOR3=-0.6
      XCOR4=XCOR3
      YCOR4=0.0
      DO 20 I=1,12
      I3=I3+1
      I4=I4+1
      BOUDC1(I3,1)=XCOR3+FLOAT(I)*DX3
      BOUDC1(I3,2)=YCOR3
      BOUDC1(I4,1)=XCOR4+FLOAT(I)*DX4
      BOUDC1(I4,2)=YCOR4
   20 CONTINUE
      J5=18
      DY5=0.125
      XCOR5L=-3.0
      YCOR5L=-0.6
      XCOR5R=1.0
      YCOR5R=YCOR5L
      DO 30 I=1,4
      J5=J5+2
      BOUDC1(J5,1)=XCOR5L
      BOUDC1(J5,2)=YCOR5L+FLOAT(I)*DY5
      BOUDC1(J5+1,1)=XCOR5R
      BOUDC1(J5+1,2)=YCOR5R+FLOAT(I)*DY5
   30 CONTINUE
      J6=26
      DY6=0.05
      XCOR6L=-3.0
      YCOR6L=-0.1
      XCOR6R=1.0
      YCOR6R=-0.1
      DO 40 I=1,1
      J6=J6+2
      BOUDC1(J6,1)=XCOR6L
      BOUDC1(J6,2)=YCOR6L+FLOAT(I)*DY6
      BOUDC1(J6+1,1)=XCOR6R
      BOUDC1(J6+1,2)=YCOR6R+FLOAT(I)*DY6
   40 CONTINUE
      DO 37 KK=1,NBCONX
      I7=KK
   37 CONTINUE
      GO TO 700
  600 CONTINUE
      I1=0
      I2=6
      DX1=2.0/6.0
      DX2=1.0/6.0
      XCOR1=-3.0
      YCOR1=0.0
      XCOR2=-1.0
      YCOR2=0.0
      DO 110 I=1,7
      I1=I1+1
```

```
      I2=I2+1
      BOUDC2(I1,1)=XCOR1+FLOAT(I-1)*DX1
      BOUDC2(I1,2)=YCOR1
      BOUDC2(I2,1)=XCOR2+FLOAT(I-1)*DX2
      BOUDC2(I2,2)=YCOR2
  110 CONTINUE
      I3=13
      NELRF=5
      NNODRF=NELRF*2
      DL3=CONLEN/NNODRF
      DO 120 I=1,NNODRF
      I3=I3+1
      BOUDC2(I3,1)=FLOAT(I)*DL3*SINA
      BOUDC2(I3,2)=FLOAT(I)*DL3*COSA
  120 CONTINUE
      I4=23
      DL4=CONLEN/6.0
      DO 130 I=1,6
      I4=I4+1
      BOUDC2(I4,1)=(CONLEN+FLOAT(I)*DL4)*SINA
      BOUDC2(I4,2)=(CONLEN+FLOAT(I)*DL4)*COSA
  130 CONTINUE
      X2=2.0*CONLEN*SINA
      Y2=2.0*CONLEN*COSA
      X1=X2-CHPTHK*COSA
      Y1=Y2+CHPTHK*SINA
      J51=28
      J52=40
      DL51=0.5/6.0
      DL52=0.55/6.0
      XCOR5=-3.0
      YCOR51=0.0
      YCOR52=0.5
      J61=29
      J62=41
      DL61=0.3*CHPTHK/6.0
      DL62=0.7*CHPTHK/6.0
      XCOR61=X2
      YCOR61=Y2
      XCOR62=X2-6.0*DL61*COSA
      YCOR62=Y2+6.0*DL61*SINA
      DO 140 I=1,6
      J51=J51+2
      BOUDC2(J51,1)=XCOR5
      BOUDC2(J51,2)=YCOR51+FLOAT(I)*DL51
      J52=J52+2
      BOUDC2(J52,1)=XCOR5
      BOUDC2(J52,2)=YCOR52+FLOAT(I)*DL61*COSA
      J61=J61+2
      BOUDC2(J61,1)=XCOR61-FLOAT(I)*DL61*COSA
      BOUDC2(J61,2)=YCOR61+FLOAT(I)*DL61*SINA
      J62=J62+2
      BOUDC2(J62,1)=XCOR62-FLOAT(I)*DL62*COSA
      BOUDC2(J62,2)=YCOR62+FLOAT(I)*DL62*SINA
  140 CONTINUE
```

```
      CALL CURVE(BOUDC2)
      DO 36 KK=1,NBCONX
      I7=KK
   36 CONTINUE
  700 RETURN
      END
C-------------------------------------------------------
C
      SUBROUTINE INTPOL(NNX,NNY,NBCON,NTOTXY,BNOD,TRSPOL,ICALL)
      DIMENSION TRSPOL(2000),BOUDV(800,2)
      INTEGER BNOD(1000)
      COMMON/A2/BOUDC1(48,2),BOUDC2(80,2)
      COMMON/AREA2/NX,NY,NOD,NEL,NDXDIR,NDYDIR
      DIMENSION XA1(500,500),XA2(500,500)
      DIMENSION YA1(500,500),YA2(500,500)
      IF(ICALL.EQ.1) GOTO 30
      IF(ICALL.EQ.2) GOTO 40
   30 CONTINUE
      DO 31 I=1,NBCON
      BOUDV(I,1)=BOUDC1(I,1)
      BOUDV(I,2)=BOUDC1(I,2)
   31 CONTINUE
      GO TO 50
   40 CONTINUE
      DO 41 I=1,NBCON
      BOUDV(I,1)=BOUDC2(I,1)
      BOUDV(I,2)=BOUDC2(I,2)
   41 CONTINUE
   50 CONTINUE
      DO 450 I=1,NBCON
      DO 450 J=1,2
  450 CONTINUE
      DO 1551 IO1=1,NDXDIR
      K44=1
      K55=2*NX+K44
      DO 1552 JO1=1,NDYDIR
      RI1=FLOAT(IO1-1)/FLOAT(NDXDIR-1)
      RI2=FLOAT(NDXDIR-IO1)/FLOAT(NDXDIR-1)
      IF(JO1.EQ.NDYDIR) K55=NBCON
      XA1(IO1,JO1)=RI1*BOUDV(K55,1)+RI2*BOUDV(K44,1)
      YA1(IO1,JO1)=RI1*BOUDV(K55,2)+RI2*BOUDV(K44,2)
      K44=K55+1
      K55=K44+1
 1552 CONTINUE
 1551 CONTINUE
      K99=1
      K77=NDXDIR+2*(NDYDIR-2)+1
      DO 1777 I=1,NDXDIR
      DO 1778 J=1,NDYDIR
      RJ1=FLOAT(J-1)/FLOAT(NDYDIR-1)
      RJ2=FLOAT(NDYDIR-J)/FLOAT(NDYDIR-1)
      XA2(I,J)=RJ1*(BOUDV(K77,1)-XA1(I,NDYDIR))+RJ2*(BOUDV(K99,
     !1)-XA1(I,1))
      TIP=BOUDV(K77,2)-YA1(I,NDYDIR)
      TEK=BOUDV(K99,2)-YA1(I,1)
```

```
      YA2(I,J)=RJ1*TIP+RJ2*TEK
 1778 CONTINUE
      K77=K77+1
      K99=K99+1
 1777 CONTINUE
      MM2=1
      DO 2300 J=1,NDYDIR,2
      DO 2330 I=1,NDXDIR
      TRSPOL(MM2)=XA1(I,J)+XA2(I,J)
      LMM2=MM2+1
      TRSPOL(LMM2)=YA1(I,J)+YA2(I,J)
      MM2=MM2+2
 2330 CONTINUE
      MM2=MM2+2*NX+2
 2300 CONTINUE
      LTT1=2*(NDXDIR)+1
      DO 2312 J=2,NDYDIR-1,2
      DO 2310 I=1,NDXDIR,2
      TRSPOL(LTT1)=XA1(I,J)+XA2(I,J)
      MLIT=LTT1+1
      TRSPOL(MLIT)=YA1(I,J)+YA2(I,J)
      LTT1=LTT1+2
 2310 CONTINUE
      LTT1=LTT1+2*NDXDIR
 2312 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
C
      SUBROUTINE BDYNOD(NBCON,BNOD)
      INTEGER BNOD(1000)
      COMMON/AREA2/NX,NY,NOD,NEL,NDXDIR,NDYDIR
      LIV=0
      DO 1880 J4=1,NDXDIR
      LIV=LIV+1
      BNOD(LIV)=J4
      LACK=NDXDIR+2*(NDYDIR-2)+1
      MASS=LIV+LACK-1
      BNOD(MASS)=NOD-NDXDIR+J4
 1880 CONTINUE
      ICOUNT=1
      KEET=1+NDXDIR
      DO 2777 JAT=2,NDYDIR-1
      BNOD(KEET)=BNOD(KEET-1)+1
      IASA=KEET+1
      BNOD(IASA)=BNOD(KEET)+NX*ICOUNT
      ICOUNT=ICOUNT+1
      KEET=KEET+2
      IF(ICOUNT.EQ.3) ICOUNT=1
 2777 CONTINUE
      RETURN
      END
C-----------------------------------------------------------------
C
      SUBROUTINE CURVE(BOUDC2)
```

```
      DIMENSION BOUDC2(80,2),ANGLN(386)
      COMMON/A4/X0,Y0,X1,Y1,X2,Y2
      COMMON/A5/RAKEAN,DCUT,CHPTHK,CONLEN,COSA,SINA
      DO 200 I=1,386
      ANGLN(I)=0.0
  200 CONTINUE
      SLOP=COSA/SINA
      PI=3.1415927
      X0=-3.0
      Y0=1.05
      BOUDC2(52,1)=X0
      BOUDC2(52,2)=Y0
      ALFA=(Y1-1.0)/(Y0-1.0)
      YBAR=((Y0-ALFA*Y1)-SLOP*(X0-ALFA*X1))/(1.0-ALFA)
      C=(Y1-SLOP*X1-YBAR)*(Y1-1.0)
      XLEN=X1-X0
      DX=XLEN/10.0
      DX1=3.0*DX/6.0
      DX2=DX1
      DX3=3.0*DX/10.0
      DX4=DX/6.0
      X01=X0
      X02=X01+3.0*DX
      X03=X02+3.0*DX
      X04=X03+3.0*DX
      IBOUD1=52
      IBOUD2=58
      IBOUD3=64
      IBOUD4=74
      I1=358
      J1=364
      K1=370
      L1=380
C EQN:(Y-MX-YBAR)*(Y-1.0)=(Y1-MX1-YBAR)(Y1-1.0)
C SLOPE FROM EQN:DYDX=M(Y-1.0)/((Y-1.0)+(Y-MX-YBAR))
      DO 100 I=1,10
      IF(I.GT.6) GOTO 23
      XCOR1=X01+I*DX1
      R1=SLOP*XCOR1+YBAR
      C1=R1-C
      YCOR1=(R1+1+SQRT((R1+1)**2-4.0*C1))/2.0
      DYDX=SLOP*(YCOR1-1.0)
      DYDX=DYDX/((YCOR1-1.0)+(YCOR1-SLOP*XCOR1-YBAR))
      BETA=ATAN(DYDX)
      ANGLE=90.0-BETA*180.0/PI
   23 IBOUD1=IBOUD1+1
      I1=I1+1
      BOUDC2(IBOUD1,1)=XCOR1
      BOUDC2(IBOUD1,2)=YCOR1
      ANGLN(I1)=ANGLE
      XCOR2=X02+I*DX2
      R2=SLOP*XCOR2+YBAR
      C1=R2-C
      YCOR2=(R2+1+SQRT((R2+1)**2-4.0*C1))/2.0
      DYDX=SLOP*(YCOR2-1.0)
```

```
      DYDX=DYDX/((YCOR2-1.0)+(YCOR2-SLOP*XCOR2-YBAR))
      BETA=ATAN(DYDX)
      ANGLE=90.0-BETA*180.0/PI
      IBOUD2=IBOUD2+1
      J1=J1+1
      BOUDC2(IBOUD2,1)=XCOR2
      BOUDC2(IBOUD2,2)=YCOR2
      ANGLN(J1)=ANGLE
      XCOR4=X04+I*DX4
      R4=SLOP*XCOR4+YBAR
      C1=R4-C
      YCOR4=(R4+1+SQRT((R4+1)**2-4.0*C1))/2.0
      DYDX=SLOP*(YCOR4-1.0)
      DYDX=DYDX/((YCOR4-1.0)+(YCOR4-SLOP*XCOR4-YBAR))
      BETA=ATAN(DYDX)
      ANGLE=90.0-BETA*180.0/PI
      IBOUD4=IBOUD4+1
      L1=L1+1
      BOUDC2(IBOUD4,1)=XCOR4
      BOUDC2(IBOUD4,2)=YCOR4
      ANGLN(L1)=ANGLE
   23 CONTINUE
      XCOR3=X03+I*DX3
      R3=SLOP*XCOR3+YBAR
      C1=R3-C
      YCOR3=(R3+1+SQRT((R3+1)**2-4.0*C1))/2.0
      DYDX=SLOP*(YCOR3-1.0)
      DYDX=DYDX/((YCOR3-1.0)+(YCOR3-SLOP*XCOR3-YBAR))
      BETA=ATAN(DYDX)
      ANGLE=90.0-BETA*180.0/PI
      IBOUD3=IBOUD3+1
      K1=K1+1
      BOUDC2(IBOUD3,1)=XCOR3
      BOUDC2(IBOUD3,2)=YCOR3
      ANGLN(K1)=ANGLE
  100 CONTINUE
      DO 234 I=1,386
      WRITE(29,*)I,ANGLN(I)
  234 CONTINUE
      RETURN
      END
C#########################################################################################
C################################## END ##################################################
```

```
C ##########################################################################
C ################ FINITE ELEMENT ANALYSIS OF ##########################
C ################## ORTHOGONAL METAL CUTTING ##########################
C ##########################################################################
C
C MASTER FLUID
C
        IMPLICIT REAL*8(A-H,O-Z)
       COMMON/ARC1/POSGP(3),WEIGP(3)
       COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
       COMMON/ARC3/VARB1(2000),VARB2(2000)
       COMMON/ARC35/EPSN(111,9),XG(111,9),YG(111,9),VISCP(111,9)
      !,DAREAG(111,9)
        COMMON/ABC/IR
        COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
      !BSTAR,SIGYO,SPHTO,THKO,SIGMA
       DIMENSION EQRHS(2000)
        DIMENSION LBOUD(2000),LHEDV(119),PNORM(119),BOUDV(2000)
       DIMENSION GFLUM(119,119)
C
        CHARACTER *50 FNAME
        WRITE(*,*)'ENTER INPUT FILE NAME'
        READ(*,'(A50)') FNAME
       OPEN(UNIT = 20,FILE=FNAME)
       OPEN(UNIT=40,FILE='VAR.INP')
       OPEN(UNIT=24,FILE='S3V3TS.OUT')
       OPEN(UNIT=35,FILE='S3V3VTS.OUT')
       OPEN(UNIT=36,FILE='S3V3TTS.OUT')
       OPEN(UNIT=28,FILE='S3V3ETS.OUT')
       OPEN(UNIT=88,FILE='TEMPTS.DAT')
       OPEN(UNIT=90,FILE='STRAINTS.DAT')
        OPEN(UNIT=38,FILE='STRESTS.DAT')
C
C************ SET DYN. DIM. VALS
        READ(20,*)IR
       CALL DIMENS(MELEM,MFRON,MPOIN,MTOTV)
C************ READ ALL PROB. DATA
       CALL DINPUT(IAXSY,BOUDV,LBOUD,MELEM,
      !MPOIN,MTOTV,NBCON,NDOFM,NELEM,NEVAB,NGAUS,NITER,
      !NNODL,NNODP,NPOIN,NTOTV,RELAX,TOLER,
      !XFORC,YFORC)
 1176 CALL ITERAT(IAXSY,GFLUM,BOUDV,LBOUD,LHEDV,PNORM,EQRHS,
      !MELEM,MFRON,MPOIN,MTOTV,NBCON,NDOFM,
      !NELEM,NEVAB,NGAUS,NITER,NNODL,NNODP,NPOIN,NTOTV,
      !RELAX,TOLER,XFORC,YFORC)
       WRITE(28,100)
C       WRITE(*,100)
C  100 FORMAT(20X,'STRAIN RATE DISTTN'//,
C     !'KELEM',45X,'GAUSS POINTS',/,
C     !18X,'1',12X,'2',12X,'3',12X,'4',10X,'5',10X,'6',10X,'7',
C     !12X,'8',12X,'9',/)
  100  FORMAT(//'ELEM  G.PT  X-COORD    Y-COORD     STRAINRATE',
      !       /'##################################################')
       DO 200 I= 1,MELEM
         DO J=1,9
```

```
      WRITE(28,*)I,J,XG(I,J),YG(I,J),EPSN(I,J)
C  111 FORMAT(I3,2X,I2,D8.4,2X,D8.4,2X,D10.4)
        ENDDO
  200 CONTINUE
        CALL FORCE
        CALL STRATES
        CALL CHTHK
      TYPE*,'MAIN OVER'
      STOP
      END
C*******************************************************
C
      SUBROUTINE DIMENS(MELEM,MFRON,MPOIN,MTOTV)
        IMPLICIT REAL*8(A-H,O-Z)
      MELEM=111
      MFRON=119
      MPOIN=386
      MTOTV=2000
      RETURN
      END
C*******************************************************
      SUBROUTINE DINPUT(IAXSY,BOUDV,LBOUD,
     !MELEM,MPOIN,MTOTV,NBCON,NDOFM,NELEM,NEVAB,
     !NGAUS,NITER,NNODL,NNODP,NPOIN,NTOTV,RELAX,
     !TOLER,XFORC,YFORC)
        IMPLICIT REAL*8(A-H,O-Z)
C
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
      DIMENSION LBOUD(2000),BOUDV(2000)
      COMMON/ARC3/VARB1(2000),VARB2(2000)
      COMMON/ARC10/CUTVEL,DCUT,WID,RAKE,GAMMA
      COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
      COMMON/ARC13/RHO,MBOUEL
      COMMON/ARC14/IBEL(60),ISURF(60),ISIDE(60)
      COMMON/ARC30/ANGLN(386)
        COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
     !BSTAR,SIGYO,SPHTO,THKO,SIGMA
        COMMON/ABC/IR
C
      RHO=7880.0
      NDOFM=4
      NEVAB=28
      NGAUS=3
      NNODL=4
      NNODP=8
      READ(20,1000) TITLE
 1000 FORMAT(12A6)
      READ(20,*) IAXSY,NELEM,NITER,NPOIN,NRPON
      READ(20,*) NICON,NBCON,NEBCN,NNBCN
      CALL DIAGN1(NBCON,NEBCN,NELEM,NICON,NNBCN,NPOIN,NRPON)
      READ(20,*)RELAX,TOLER,XFORC,YFORC
      READ(20,*) EN,GAMMA,SIGYO
      READ(20,*) CUTVEL,DCUT,RAKE,FRIFAC,WID
      PI=3.1415927
      RAKEAN=RAKE*PI/180.0
```

```
      RAKE2=2.0*RAKEAN
      SIN2A=DSIN(RAKE2)
    COSA=DCOS(RAKEAN)
    SINA=DSIN(RAKEAN)
    BSTAR=(CUTVEL/(1.732*DCUT*GAMMA))**(1.0/EN)
    BSTAR=BSTAR/SIGY0
    READ(20,*) H,THK0,THKTOL,RHO,SPHTO
    READ(20,*) TAMB,TWP
    ALPHA=THK0/(RHO*SPHTO)
    PECLET=CUTVEL*DCUT/ALPHA
    TREF=SIGY0/(RHO*SPHTO)
    HSTAR=H*DCUT/THK0
    TFSTAR=TAMB/TREF
    CONRAT=THK0/(THKTOL+THK0)
      SIGMA=SIGY0/(RHO*CUTVEL**2)
      WRITE(35,355)DCUT,CUTVEL,RAKE,H,THK0,SPHTO,RHO,FRIFAC
355 FORMAT(//3X,' DATA USED',
   !         /2X,'##################',
   !        //4X,'DEPTH OF CUT :',F12.8,3X,'CUTTING VEL.:',D12.4,
   !         /4X,'RAKE ANGLE   :',D12.4,3X,'HT.TR.COEFF.:',D12.4,
   !         /4X,'THERM.COND.  :',D12.4,3X,'SP.HEAT      :',D12.4
   !         /4X,'DENSITY      :',D12.4,3X,'FRIC. FAC.   :',D12.4,
   !         /3X,'##############################################')
    DO 10 IPOIN=1,NPOIN
    DO 10 IDIME=1,2
    COORD(IPOIN,IDIME)=0.0
 10 CONTINUE
    DO 20 IPOIN=1,NRPON
    JPOIN=IPOIN
    READ(20,*) JPOIN,(COORD(IPOIN,IDIME),IDIME=1,2)
 20 CONTINUE
    READ(20,*) SCALE
    DO 25 IDIME=1,2
    DO 25 IPOIN=1,NRPON
    COORD(IPOIN,IDIME)=COORD(IPOIN,IDIME)*SCALE/DCUT
 25 CONTINUE
    DO 30 IELEM=1,NELEM
    JELEM=IELEM
    READ(20,*) JELEM,(LNODS(JELEM,INODP),INODP=1,NNODP)
 30 CONTINUE
    READ(20,*) MSURF
    READ(20,*) MELS1,MELS2,MELS3,MELS4,MELS5,MELS6,MELS7,MELS8
    MBOUEL=MELS1+MELS2+MELS3+MELS4+MELS5+MELS6+MELS7+MELS8
    DO 9 II=1,MBOUEL
    JJ=II
    READ(20,*) JJ,ISURF(II),IBEL(II),ISIDE(II)
  9 CONTINUE
C   SET UP A VECTOR GIVING D.O.F AT EACH NODE
    ITEMP=NNODP-1
    DO 40 IELEM=1,NELEM
    DO 40 INODP=1,ITEMP,2
    NODFM(LNODS(IELEM,INODP))=NDOFM
    NODFM(LNODS(IELEM,INODP+1))=NDOFM-1
 40 CONTINUE
C
```

```
C   INTERPOLE MIDSIDE NODE WHERE NOT KNOWN
        GO TO 71
        DO 70 IELEM=1,NELEM
        DO 60 INODP=2,NNODP,2
        IPOIN=LNODS(IELEM,INODP)
        TEMPY=DABS(COORD(IPOIN,1))+DABS(COORD(IPOIN,2))
        IF(TEMPY.NE.0.0) GOTO 60
        JPOIN=LNODS(IELEM,INODP-1)
        KNODP=INODP+1
        IF(KNODP.GT.NNODP) KNODP=1
        KPOIN=LNODS(IELEM,KNODP)
        DO 50 IDIME=1,2
        COORD(IPOIN,IDIME)=(COORD(JPOIN,IDIME)+COORD(KPOIN,IDIME))*0.5
   50 CONTINUE
   60 CONTINUE
   70 CONTINUE
   71 CONTINUE
C
C   SETUP A VECTOR WITH FIRST D.O.F AT EACH NODE
        NADFM(1)=1
        DO 80 IPOIN=2,NPOIN
        NADFM(IPOIN)=NADFM(IPOIN-1)+NODFM(IPOIN-1)
   80 CONTINUE
        NTOTV=NADFM(NPOIN)+NODFM(NPOIN)-1
C *** INITIALISE REMAINING ARRAYS
        DO 90 ITOTV=1,NTOTV
        LBOUD(ITOTV)=0
        BOUDV(ITOTV)=0.0
        VARB1(ITOTV)=0.0
        VARB2(ITOTV)=0.0
   90 CONTINUE
C
          IF(IR.EQ.1)THEN
          DO JTOTV=1,NTOTV
          READ(40,*)IPOIN,IDOFM,VARB2(JTOTV)
          ENDDO
          DO IICON=1,NICON
          READ(20,*)
          ENDDO
          GOTO 110
          ENDIF
C
C ****READ INITIAL CODN.S IF ANY
        IF(NICON.EQ.0) GO TO 110
        DO 100 IICON=1,NICON
        READ(20,*) IPOIN,IDOFM,TEMPY
        IF(IDOFM.GT.1) IDOFM=NODFM(IPOIN)
        JTOTV=NADFM(IPOIN)+IDOFM-1
        VARB1(JTOTV)=TEMPY
        VARB2(JTOTV)=TEMPY
  100 CONTINUE
  110 CONTINUE
C
C CHECK NODAL CONN.S & COORDINATES\
        CALL DIAGN2(MELEM,MPOIN,NELEM,NICON,
```

```
     !NNODP,NPOIN,NTOTV)
C
      DO 175 IBCON=1,NBCON
        READ(20,*) IPOIN,IDOFM,BVALU
        IF((IPOIN.EQ.1).AND.(IDOFM.EQ.2))THEN
        WRITE(35,799) BVALU
  799 FORMAT(/5X,'PRESSURE AT NODE NO. 1 IS ',D8.2,' OF SIGMAY')
        ENDIF
        IF(IDOFM.EQ.4)BVALU=BVALU/TREF
      ITOTV=NADFM(IPOIN)+IDOFM-1
      IF(NODFM(IPOIN).EQ.4) GOTO 238
      IF((IDOFM+1).GT.NODFM(IPOIN)) ITOTV=ITOTV-1
  238 LBOUD(ITOTV)=1
      BOUDV(ITOTV)=BVALU
      GO TO(140,150,160,165) IDOFM
  140 CONTINUE
      GOTO 170
  150 CONTINUE
        GOTO 170
      IF(NODFM(IPOIN).NE.(IDOFM+1)) GOTO 170
        STOP
  160 CONTINUE
      GOTO 170
  165 CONTINUE
      JTOTV=ITOTV
  170 CONTINUE
  175 CONTINUE
      DO 990 ITOTV=1,NTOTV
      JTOTV=ITOTV
  990 CONTINUE
        DO 999 I=1,MPOIN
        J=I
        READ(20,*)J, ANGLN(I)
  999 CONTINUE
      DO 400 KKL=1,MPOIN
      LL=KKL
  400 CONTINUE
      RETURN
      END
C ***************************************************
C
      SUBROUTINE DIAGN1(NBCON,NEBCN,NELEM,NICON,NNBCN,NPOIN,NRPON)
        IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION NECHO(150),NEROR(8)
C    SCRUTINY OF CONTROL DATA
      DO 10 IEROR=1,8
   10 NEROR(IEROR)=0
C  SCRUTINISE CONTROL DATA & PRINT MESSAGE
      IF(NPOIN.LE.0)                          NEROR(1)=1
      IF(NRPON.LE.0.OR.NRPON.GT.NPOIN)NEROR(2)=2
      IF(NELEM.LE.0)                          NEROR(3)=3
      IF(NPOIN.GT.NELEM*8)                    NEROR(4)=4
      IF(NBCON.GT.NPOIN*4)                    NEROR(5)=5
      IF(NICON.GT.NPOIN*4)                    NEROR(6)=6
      IF(NEBCN.GT.NELEM)                      NEROR(7)=7
```

```
      IF(NNBCN.GT.NPOIN*4)              NEROR(8)=8
      JEROR=0
      DO 20 IEROR=1,8
      IF(NEROR(IEROR).EQ.0)GOTO 20
      JEROR=1
C      WRITE(24,2000)IEROR
C 2000 FORMAT(/,33H CONTROL PARAMETER ERROR****ERROR,I5)
   20 CONTINUE
      IF(JEROR.EQ.0) RETURN
C      WRITE(24,2010)
C 2010 FORMAT(/10X,37HDATA FOLLOWING ERROR IN CONTROL CARDS/)
   30 CONTINUE
      READ(20,1000) NECHO
 1000 FORMAT(150A1)
      WRITE(24,2020) NECHO
 2020 FORMAT(10X,80A1)
      GOTO 30
      RETURN
      END
C ***************************************************************************
C
      SUBROUTINE DIAGN2(MELEM,MPOIN,NELEM,NICON
     !,NNODP,NPOIN,NTOTV)
        IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION NECHO(150),NEROR(13)
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
      DO 10 IEROR=1,13
      NEROR(IEROR)=0
   10 CONTINUE
C CHECK PHY. PROPS
      IF(DENS.LE.0.0.OR.VISCY.LE.0.0)NEROR(9)=1
C CHECK FOR IDENT. COORDS
      DO 40 IPOIN=2,NPOIN
      JPOIN=IPOIN-1
      DO 30 KPOIN=1,JPOIN
      IF(COORD(IPOIN,1).NE.COORD(KPOIN,1).OR.COORD(IPOIN,2).NE.
     !COORD(KPOIN,2)) GOTO 20
      NEROR(10)=1
   20 CONTINUE
   30 CONTINUE
   40 CONTINUE
C CHECK ELEMENT NODAL NUMBERS
C A-REAPITATION OF NOS.
C B-NO. OUTSIDE PERM. BOUND
      DO 50 IPOIN=1,NPOIN
      DO 50 IELEM=1,NELEM
      LCONT=0
      DO 50 INODE=1,NNODP
      IF(LNODS(IELEM,INODE).NE.IPOIN) GOTO 50
      LCONT=LCONT+1
      IF(LCONT.GT.1)NEROR(11)=1
   50 CONTINUE
      DO 60 IELEM=1,NELEM
      DO 60 INODE=1,NNODP
      IF(LNODS(IELEM,INODE).LE.0.OR.LNODS(IELEM,INODE).GT.NPOIN)
```

```
      !NEROR(12)=1
   60 CONTINUE
C CHECK NO OF INITIAL CONDNS
      IF(NICON.GT.NTOTV)NEROR(13)=1
      JEROR=0
      DO 70 IEROR=10,13
      IF(NEROR(IEROR).EQ.0)GO TO 70
      JEROR=1
      WRITE(24,2000) IEROR
 2000 FORMAT(/10X,10H DATA ERROR,I5)
   70 CONTINUE
      IF(JEROR.EQ.0) RETURN
      WRITE(24,2010)
 2010 FORMAT(/10X,14HREMAINING DATA/)
   80 CONTINUE
      READ(20,1000)NECHO
 1000 FORMAT(150A1)
C      WRITE(25,2020) ECHO
C 2020 FORMAT(/10X,80A1)
      RETURN
      END
C*****************************************************************
C
      SUBROUTINE ITERAT(IAXSY,GFLUM,BOUDV,LBOUD,LHEDV,PNORM,EQRHS
     !,MELEM,MFRON,MPOIN,MTOTV,NBCON,NDOFM,
     !NELEM,NEVAB,NGAUS,NITER,NNODL,NNODP,NPOIN,NTOTV,
     !RELAX,TOLER,XFORC,YFORC)
        IMPLICIT REAL*8(A-H,O-Z)
C EXT. SUBROUTINES
C PRESCR
C FRONTS
C WRITER
C TOLREL
      COMMON/ARC1/POSGP(3),WEIGP(3)
      COMMON/ARC2/LNODS(111,8),COORDS(386,2),NADFM(386),NODFM(386)
      COMMON/ARC3/VARB1(2000),VARB2(2000)
      COMMON/ARC35/EPSN(111,9),XG(111,9),YG(111,9),VISCP(111,9)
     !,DAREAG(111,9)
      DIMENSION EQRHS(2000)
      DIMENSION LBOUD(2000),LHEDV(119),PNORM(119),BOUDV(2000)
      DIMENSION GFLUM(119,119)
C SETUP ITER COUNTER
      IITER=0
      DO 1 I=1,MELEM
      DO 1 J=1,9
      EPSN(I,J)=0.0
    1 CONTINUE
   10 CONTINUE
      IITER=IITER+1
      WRITE(35,*) '  IITER=',IITER
      DO 20 ITOTV=1,NTOTV
      EQRHS(ITOTV)=0.0
   20 CONTINUE
C CALL FRONTS TO SETUP AND SOLVE GOVERNIG EQNS.
      DO 50 I=1,MFRON
```

```
      DO 50 J=1,MFRON
      GFLUM(I,J)=0.0
   50 CONTINUE
      KOUNT=1
      NESPBC=0
      CALL FRONTS(IAXSY,IITER,MELEM,MFRON,MPOIN,MTOTV,NBCON,NELEM,
     !NEVAB,NNODL,NNODP,NPOIN,NTOTV,XFORC,YFORC,NGAUS,NESPBC,GFLUM,
     !BOUDV,LBOUD,LHEDV,PNORM,EQRHS)
      KOUNT=KOUNT+1
C  CALL WRITER TO O/P ITER. RESULTS
C       CALL WRITER(IITER,MPOIN,MTOTV,NDOFM,NPOIN,MELEM)
      KOUNT=KOUNT+1
C CALL TOLREL TO CHECK CONVERGENCE AND RELAX VALUE IF NOT
      CALL TOLREL(IITER,MTOTV,NCONV,NTOTV,RELAX,TOLER,VARB1,VARB2
     !,NADFM,NODFM,MPOIN)
      KOUNT=KOUNT+1
C  RETURN TO MASTER NO. IF ITERATIONS EXCEED MAXIMUM
      IF(NCONV.EQ.1) GOTO 12
      IF(IITER.LT.NITER) GOTO 10
      WRITE(*,2000)
      WRITE(35,2000)
 2000 FORMAT(//32H SOLUTION HAS FAILED TO CONVERGE)
   12 CALL WRITER(IITER,MPOIN,MTOTV,NDOFM,NPOIN,MELEM)
      RETURN
      END
C*****************************************************************
      SUBROUTINE WRITER(IITER,MPOIN,MTOTV,NDOFM,NPOIN,MELEM)
        IMPLICIT REAL*8(A-H,O-Z)
        DIMENSION VARBX(2000)
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
      COMMON/ARC3/VARB1(2000),VARB2(2000)
      COMMON/ARC10/CUTVEL,DCUT,RAKE
        COMMON/ABC/IR
        COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
     !BSTAR,SIGYO,SPHTO,THKO,SIGMA
      WRITE(*,111) IITER
  111 FORMAT(1X,'IITER=',I5)
      WRITE(35,2000) IITER
 2000 FORMAT(//29H RESULTS FOR ITERATION NUMBER,I2,
     !//14X,4(3HNEW,7X),7X,4(3HOLD,7X)/,
     !7H NODE ,2(4X,42HU-VELOCITY PRESSURE V-VELOCITY TEMPARATURE))
      DO 20 IPOIN=1,MPOIN
      IODFM=NODFM(IPOIN)
      IADFM=NADFM(IPOIN)
      IF(IODFM.EQ.NDOFM) GOTO 10
      WRITE(35,2010)IPOIN,(VARB1(IADFM+JODFM-1),JODFM=1,IODFM),
     !(VARB2(IADFM+JODFM-1),JODFM=1,IODFM)
        DO JODFM=1,IODFM
        JK=JODFM
        IF(JK.GT.1)JK=JK+1
        KK=IADFM+JODFM-1
        WRITE(40,*)IPOIN,JK,VARB1(KK)
        IF(JK.EQ.4)THEN
        VARBX(KK)=VARB1(KK)*TREF
        WRITE(88,*)IPOIN,VARBX(KK)
```

```
        ENDIF
        ENDDO
      GOTO 20
   10 CONTINUE
      WRITE(35,2020)IPOIN,(VARB1(IADFM+JODFM-1),JODFM=1,IODFM),
     !(VARB2(IADFM+JODFM-1),JODFM=1,IODFM)
        DO JODFM=1,IODFM
        JK=JODFM
        KK=IADFM+JODFM-1
        WRITE(40,*)IPOIN,JK,VARB1(KK)
        IF(JK.EQ.4)THEN
        VARBX(KK)=VARB1(KK)*TREF
        WRITE(88,*)IPOIN,VARBX(KK)
        ENDIF
        ENDDO
   20 CONTINUE
 2010 FORMAT(I6,4X,2(E10.3,10X,2E10.3,6X))
 2020 FORMAT(I6,4X,2(4E10.3,6X))
      RETURN
      END
C*********************************************************************
C
      SUBROUTINE TOLREL(IITER,MTOTV,NCONV,NTOTV,RELAX,TOLER,
     !VARB1,VARB2,NADFM,NODFM,MPOIN)
        IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION VARB1(2000),VARB2(2000),NADFM(386),NODFM(386)
C TO CHECK IF SOLN. HAS CONVERGED
      NDOFM=4
      DO 208 IPOIN=1,MPOIN
      IODFM=NODFM(IPOIN)
      IADFM=NADFM(IPOIN)
      IF(IODFM.EQ.NDOFM) GOTO 108
      GOTO 208
  108 CONTINUE
  208 CONTINUE
      CANLA=0.0
      NCONV=1
      UMAX=VARB1(1)
      PMAX=VARB1(2)
      VMAX=VARB1(3)
      TMAX=VARB1(4)
      DO 10 IPOIN=1,MPOIN
      IODFM=NODFM(IPOIN)
      IADFM=NADFM(IPOIN)
      IF(IODFM.EQ.3) GOTO 16
      IF(DABS(VARB1(IADFM)).GT.DABS(UMAX)) UMAX=VARB1(IADFM)
      IF(DABS(VARB1(IADFM+1)).GT.DABS(PMAX)) PMAX=VARB1(IADFM+1)
      IF(DABS(VARB1(IADFM+2)).GT.DABS(VMAX)) VMAX=VARB1(IADFM+2)
      IF(DABS(VARB1(IADFM+3)).GT.DABS(TMAX)) TMAX=VARB1(IADFM+3)
      GOTO 100
   16 IF(DABS(VARB1(IADFM)).GT.DABS(UMAX)) UMAX=VARB1(IADFM)
      IF(DABS(VARB1(IADFM+1)).GT.DABS(VMAX)) VMAX=VARB1(IADFM+1)
      IF(DABS(VARB1(IADFM+2)).GT.DABS(TMAX)) TMAX=VARB1(IADFM+2)
  100 CONTINUE
   10 CONTINUE
```

```
      UCUT=0.1*UMAX
      PCUT=0.1*PMAX
      VCUT=0.1*VMAX
      TCUT=0.1*TMAX
      DO 18 IPOIN=1,MPOIN
      IODFM=NODFM(IPOIN)
      IADFM=NADFM(IPOIN)
      ITOTU=IADFM
      IF(IODFM.EQ.3) GOTO 19
      ITOTP=IADFM+1
      ITOTV=ITOTU+2
      ITOTT=ITOTU+3
      GOTO 34
   19 ITOTV=ITOTU+1
      ITOTT=ITOTU+2
   34 IF(DABS(VARB1(ITOTU)).LT.DABS(UCUT)) GOTO 60
      IF(DABS(VARB1(ITOTU)).LT.0.01) GOTO 60
      CANGE=DABS((VARB1(ITOTU)-VARB2(ITOTU))/VARB1(ITOTU))
      IF(CANGE.GT.TOLER)NCONV=0
      IF(CANLA.GT.CANGE) GOTO 60
      CANLA=CANGE
      LTOLA=ITOTU
   60 CONTINUE
      IF(IODFM.EQ.3) GOTO 70
      IF(DABS(VARB1(ITOTP)).LT.DABS(PCUT)) GOTO 70
      IF(DABS(VARB1(ITOTP)).LT.0.01)GOTO 70
      CANGE=DABS((VARB1(ITOTP)-VARB2(ITOTP))/VARB1(ITOTP))
      IF(CANGE.GT.TOLER)NCONV=0
      IF(CANLA.GT.CANGE)GOTO 70
      CANLA=CANGE
      LTOLA=ITOTP
   70 CONTINUE
      IF(DABS(VARB1(ITOTV)).LT.DABS(VCUT)) GOTO 80
      IF(DABS(VARB1(ITOTV)).LT.0.01) GOTO 80
      CANGE=DABS((VARB1(ITOTV)-VARB2(ITOTV))/VARB1(ITOTV))
      IF(CANGE.GT.TOLER)NCONV=0
      IF(CANLA.GT.CANGE)GOTO 80
      CANLA=CANGE
      LTOLA=ITOTV
   80 CONTINUE
      IF(DABS(VARB1(ITOTT)).LT.DABS(TCUT)) GOTO 18
      IF(DABS(VARB1(ITOTT)).LT.0.01)GOTO 18
      CANGE=DABS((VARB1(ITOTT)-VARB2(ITOTT))/VARB1(ITOTT))
      IF(CANGE.GT.TOLER)NCONV=0
      IF(CANLA.GT.CANGE)GOTO 18
      CANLA=CANGE
      LTOLA=ITOTT
   18 CONTINUE
      WRITE(*,2000)LTOLA,CANLA
      WRITE(35,2000)LTOLA ,CANLA
 2000 FORMAT(/37X,' LARGEST CHANGE OCCUR AT DOF NO ',I5,/,
     !10X,' CHANGE =',F10.4)
      IF(NCONV.EQ.0) GOTO 20
      WRITE(*,2010)IITER
      WRITE(35,2010)IITER
```

```
 2010 FORMAT(//45H SOLUTION HAS CONVERGED TO REQUIRED TOLERANCE/,
     !3X,'IN',3X,I5,3X,'ITERATIONS')
      RETURN
   20 CONTINUE
      IF(IITER.EQ.1)GOTO 40
C
C RELAX VAR. FOR NEXT ITERATIONS
      DO 222 IPOIN=1,MPOIN
      IODFM=NODFM(IPOIN)
      IADFM=NADFM(IPOIN)
      DO 21 JODFM=1,IODFM
        REL=RELAX
        IF((IODFM.EQ.4).AND.(JODFM.EQ.2)) REL=1.0
      TEMPY1=VARB1(IADFM+JODFM-1)
      TEMPY2=VARB2(IADFM+JODFM-1)
      VARB2(IADFM+JODFM-1)=TEMPY2+REL*(TEMPY1-TEMPY2)
   21 CONTINUE
  222 CONTINUE
      GO TO 225
   40 CONTINUE
      DO 50 ITOTV=1,NTOTV
      VARB2(ITOTV)=VARB1(ITOTV)
   50 CONTINUE
  225 CONTINUE
      RETURN
      END
C*****************************************************************
C
      SUBROUTINE FRONTS(IAXSY,IITER,MELEM,MFRON,MPOIN,MTOTV,NBCON,NE
     !LEM,NEVAB,NNODL,NNODP,NPOIN,NTOTV,XFORC,YFORC,NGAUS,NESPBC,GFLU
     !M,BOUDV,LBOUD,LHEDV,PNORM,EQRHS)
        IMPLICIT REAL*8(A-H,O-Z)
C*******************************
C     EXTTERNAL SUBROUTINES
C     MATRIX
C*******************************
      DIMENSION LBOUD(2000),LHEDV(119),PNORM(119),BOUDV(2000)
      DIMENSION EQRHS(2000)
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
        COMMON/ARC3/VARB1(2000),VARB2(2000)
      DIMENSION LOCEL(28),NDEST(28),FLUMX(28,28),GFLUM(119,119)
      OPEN(UNIT=25,FORM='UNFORMATTED',FILE='ASDTS.OUT')
        IF(KELEM.EQ.NELEM) GOTO 67
   67 CONTINUE
      IF(IITER.GT.1)GOTO 40
C ON FIRST ITERATION ONLY FIND LAST APPEARNCE OF EACH NODE
      DO 30 IPOIN=1,NPOIN
      LASTE=0
      DO 20 IELEM=1,NELEM
      DO 10 INODP=1,NNODP
      IF(LNODS(IELEM,INODP).NE.IPOIN) GO TO 10
      LASTE=IELEM
      LASTN=INODP
      GOTO 20
   10 CONTINUE
```

```
   20 CONTINUE
        LNODS(LASTE,LASTN)=-IPOIN
   30 CONTINUE
   40 CONTINUE
        REWIND 25
      NCRIT=MFRON-NEVAB
      NFRON=0
      DO 50 IFRON=1,MFRON
      DO 50 JFRON=1,MFRON
      GFLUM(IFRON,JFRON)=0.0
   50 CONTINUE
      KELEM=0
C START ASSEMBLY BY FORMING ELEMENTAL MATRICES
   60 CONTINUE
      KELEM=KELEM+1
      CALL MATRIX(IAXSY,MELEM,EQRHS,FLUMX,MPOIN
     !,MTOTV,NEVAB,NNODL,NNODP,XFORC,YFORC
     !,KELEM,NELEM,NGAUS,IITER)
C       IF(KELEM.EQ.NELEM)GOTO 3770
C 3770 RETURN
      KEVAB=0
C  CREATE GLOBAL DOF ARRAY FOR EACH LOCAL ELEMENT DOF
      DO 70 INODP=1,NNODP
      KPOIN=LNODS(KELEM,INODP)
      IADFM=NADFM(IABS(KPOIN))
      LODFM=NODFM(IABS(KPOIN))
      DO 70 IODFM=1,LODFM
      KEVAB=KEVAB+1
      LOCEL(KEVAB)=IADFM+IODFM-1
      IF(KPOIN.LT.0)LOCEL(KEVAB)=-LOCEL(KEVAB)
   70 CONTINUE
C FIT EACH DOF INTO THE FRONT WIDTH EXTENDING IF NECESARY
      DO 120 IEVAB=1,NEVAB
      KTOTV=LOCEL(IEVAB)
      IF(NFRON.EQ.0)GOTO 90
      DO 80 IFRON=1,NFRON
      KFRON=IFRON
      IF(IABS(KTOTV).EQ.IABS(LHEDV(KFRON)))GOTO 110
   80 CONTINUE
   90 CONTINUE
      NFRON=NFRON+1
      IF (NFRON.LE.MFRON)GOTO 100
      WRITE(*,2000)
 2000 FORMAT(//3X,'PROGRAM HALTED FRONT WIDTH IS TOO SMALL')
      STOP
  100 CONTINUE
      NDEST(IEVAB)=NFRON
      LHEDV(NFRON)=KTOTV
      GOTO 120
  110 CONTINUE
      NDEST(IEVAB)=KFRON
      LHEDV(KFRON)=KTOTV
  120 CONTINUE
C ASSEMBLE NEW ELEMENNT IN TO GRAND FLUID MATRIX
      DO 130 IEVAB=1,NEVAB
```

```
      IFRON=NDEST(IEVAB)
      DO 130 JEVAB=1,NEVAB
      JFRON=NDEST(JEVAB)
      GFLUM(JFRON,IFRON)=GFLUM(JFRON,IFRON)+FLUMX(JEVAB,IEVAB)
  130 CONTINUE
      IF(NFRON.LT.NCRIT.AND.KELEM.LT.NELEM) GOTO 60
  140 CONTINUE
      NFSUM=0
      PIVOT=0.0
C CHECK LAST APPEARANCE OF EACH DOF PROCESS BOUN. CODNS.
      DO 170 IFRON=1,NFRON
      IF(LHEDV(IFRON).GE.0)GOTO 170
      NFSUM=1
      IF(LBOUD(IABS(LHEDV(IFRON))).NE.1)GOTO 160
      KTOTV=IABS(LHEDV(IFRON))
      LBOUD(KTOTV)=-1
      EQRHS(KTOTV)=BOUDV(KTOTV)
      DO 150 LFRON=1,NFRON
      GFLUM(IFRON,LFRON)=0.0
  150 CONTINUE
      GFLUM(IFRON,IFRON)=1.0
  160 CONTINUE
C SEARCH FOR LARGEST PIVOTAL VAL
      PIVOG=GFLUM(IFRON,IFRON)
      IF(DABS(PIVOG).LT.DABS(PIVOT))GOTO 170
      PIVOT=PIVOG
      LPIVT=IFRON
  170 CONTINUE
      IF(NFSUM.EQ.0)GOTO 60
      KTOTV=IABS(LHEDV(LPIVT))
      IF(DABS(PIVOT).GT.1.0E-25)GOTO 180
C       IF(DABS(PIVOT).GE..0E+00)GOTO 180
      WRITE(*,2010)KTOTV,PIVOT
 2010 FORMAT(///'PROGRAM HALTED ILL-CONDITIONING,'///'D.O.FREEDOM'
     !,I4,/'PIVOT VALUE',E9.2)
      STOP
  180 CONTINUE
C************ NORMALISE PIVOTAL EQN
      DO 190 IFRON=1,NFRON
      PNORM(IFRON)=GFLUM(LPIVT,IFRON)/PIVOT
  190 CONTINUE
      RHSID=EQRHS(KTOTV)/PIVOT
      EQRHS(KTOTV)=RHSID
C ***ELIMINATION OF PIVOTAL EQN.REDUCING FRONT WIDTH
      IF(LPIVT.EQ.1)GOTO 250
      DO 240 IFRON=1,LPIVT-1
      FACOR=GFLUM(IFRON,LPIVT)
      IF(DABS(FACOR).LT.1.0E-30) GOTO 210
      DO 200 JFRON=1,LPIVT-1
      GFLUM(IFRON,JFRON)=GFLUM(IFRON,JFRON)-FACOR*PNORM(JFRON)
  200 CONTINUE
  210 CONTINUE
      IF(LPIVT.EQ.NFRON)GOTO 230
      DO 220 JFRON=LPIVT+1,NFRON
      GFLUM(IFRON,JFRON-1)=GFLUM(IFRON,JFRON)-FACOR*PNORM(JFRON)
```

```
  220 CONTINUE
  230 CONTINUE
      ITOTV=IABS(LHEDV(IFRON))
      EQRHS(ITOTV)=EQRHS(ITOTV)-FACOR*RHSID
  240 CONTINUE
  250 CONTINUE
      IF(LPIVT.EQ.NFRON)GOTO 300
      DO 290 IFRON=LPIVT+1,NFRON
      FACOR=GFLUM(IFRON,LPIVT)
      IF(LPIVT.EQ.1)GOTO 270
      DO 260 JFRON=1,LPIVT-1
      GFLUM(IFRON-1,JFRON)=GFLUM(IFRON,JFRON)-FACOR*PNORM(JFRON)
  260 CONTINUE
  270 CONTINUE
      DO 280 JFRON=LPIVT+1,NFRON
      GFLUM(IFRON-1,JFRON-1)=GFLUM(IFRON,JFRON)-FACOR*PNORM(JFRON)
  280 CONTINUE
      ITOTV=IABS(LHEDV(IFRON))
      EQRHS(ITOTV)=EQRHS(ITOTV)-FACOR*RHSID
  290 CONTINUE
  300 CONTINUE
C***WRITE NONFIXED PIVOTAL EQN.
      IF(LBOUD(KTOTV).NE.0)GOTO 310
      WRITE(25)NFRON,LPIVT,(LHEDV(IFRON),PNORM(IFRON),IFRON=1,NFRON)
  310 CONTINUE
      DO 320 IFRON=1,NFRON
      GFLUM(IFRON,NFRON)=0.0
      GFLUM(NFRON,IFRON)=0.0
  320 CONTINUE
      IF(LPIVT.EQ.NFRON)GOTO 340
      DO 330 IFRON=LPIVT,NFRON-1
      LHEDV(IFRON)=LHEDV(IFRON+1)
  330 CONTINUE
  340 CONTINUE
      NFRON=NFRON-1
C****ASSEMBLE ELIMINATE OR BACK SUBSTITUTE
      IF(NFRON.GT.NCRIT)GOTO 140
      IF(KELEM.LT.NELEM)GOTO 60
      IF(NFRON.GT.0)GOTO 140
C************BACK SUBSTITUTION
      DO 350 ITOTV=1,NTOTV
      VARB1(ITOTV)=ROUDV(ITOTV)
      LBOUD(ITOTV)=-LBOUD(ITOTV)
  350 CONTINUE
      DO 370 ITOTV=1,NTOTV-NBCON
      BACKSPACE 25
      READ(25)NFRON,LPIVT,(LHEDV(IFRON),PNORM(IFRON),IFRON=1,NFRON)
      KTOTV=IABS(LHEDV(LPIVT))
      TEMPR=0.0
      PNORM(LPIVT)=0.0
      DO 360 IFRON=1,NFRON
      TEMPR=TEMPR-PNORM(IFRON)*VARB1(IABS(LHEDV(IFRON)))
  360 CONTINUE
      VARB1(KTOTV)=EQRHS(KTOTV)+TEMPR
      BACKSPACE 25
```

```
  370 CONTINUE
      RETURN
      END
C*****************************************************************************
C
      SUBROUTINE MATRIX(IAXSY,MELEM,EQRHS,FLUMX,
     !MPOIN,MTOTV,NEVAB,NNODL,NNODP,XFORC,
     !YFORC,KELEM,NELEM,NGAUS,IITER)
        IMPLICIT REAL*8(A-H,O-Z)
C***INSERTED THE DUMMY VARIABLE "NSIDE" ABOVE***********
      DIMENSION CARTL(2,4),CARTP(2,8),ERHSU(8),ERHSV(8),FLUMX(28,28)
      DIMENSION ERHST(8),VISC(9),SHAPX(8)
      DIMENSION SHAPL(4),SHAPP(8),AREAW(9),DERX(2,8),DERIV(2,8)
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
      DIMENSION EQRHS(2000)
      COMMON/AREA3/CARPG(2,72),SHAPG(72),CARLG(2,36),SHALG(36)
        COMMON/ARC1/POSGP(3),WEIGP(3)
        COMMON/ARC3/VARB1(2000),VARB2(2000)
C       COMMON/ARC4/AREAW(9)
      COMMON/ARC10/CUTVEL,DCUT,RAKE
      COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
      COMMON/ARC13/RHO,MBOUEL
      COMMON/ARC14/IBEL(60),ISURF(60),ISIDE(60)
      COMMON/ARC35/EPSN(111,9),XG(111,9),YG(111,9),VISCP(111,9)
     !,DAREAG(111,9)
        COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
     !BSTAR,SIGYO,SPHTO,THKO,SIGMA
      CALL DRIVES(COORD,LNODS,MELEM,MPOIN,NGAUS,NELEM,NNODL,NNODP,
     !KELEM,AREAW)
C
C**** INITIALISE ARRAYS
      DO 10 INODP=1,NNODP
      ERHSU(INODP)=0.0
      ERHSV(INODP)=0.0
      ERHST(INODP)=0.0
   10 CONTINUE
      DO 20 IEVAB=1,NEVAB
      DO 20 JEVAB=1,NEVAB
      FLUMX(IEVAB,JEVAB)=0.0
   20 CONTINUE
C*********************************
C
C  LOOP TO CARRY OUT GUASS INTEGRATIONS
        IGAUS=0
        DO 999 IG=1,3
        DO 999 JG=1,3
C       DO 100 IGAUS=1,9
        IGAUS=IGAUS+1
        XEQIV=POSGP(IG)
        YEQIV=POSGP(JG)
        CALL SHAPE8(DERIV,SHAPX,XEQIV,YEQIV)
        TEMPR=0.
C
        XG(KELEM,IGAUS)=0.
        YG(KELEM,IGAUS)=0.
```

```
          DO 222 INODP=1,8
          KPOIN=IABS(LNODS(KELEM,INODP))
          ITOTU=NADFM(KPOIN)
          ITOTT=ITOTU+NODFM(KPOIN)-1
          TEMPR=TEMPR+VARB2(ITOTT)*SHAPX(INODP)
C FINDIND GLOBAL POS. OF G.PTS
          XG(KELEM,IGAUS)=XG(KELEM,IGAUS)+(SHAPX(INODP)*COORD(KPOIN,1))
     !*DCUT*1000.
          YG(KELEM,IGAUS)=YG(KELEM,IGAUS)+(SHAPX(INODP)*COORD(KPOIN,2))
     !*DCUT*1000.
222     CONTINUE
C
          CALL PROP(TEMPR,SIGSTAR,CPSTAR,THKSTAR,SIGMAY)
C
          DO KINP=1,8
          KPOIN=IABS(LNODS(KELEM,KINP))
          ITOTU=NADFM(KPOIN)
          ITOTT=ITOTU+NODFM(KPOIN)-1
        KIGS=IGAUS
        DAREA=AREAW(KIGS)
          DAREAG(KELEM,KIGS)=DAREA
        DO 30 INODP=1,NNODP
        SHAPP(INODP)=SHAPG(NNODP*(IGAUS-1)+INODP)
        DO 30 IDIME=1,2
        CARTP(IDIME,INODP)=CARPG(IDIME,NNODP*(IGAUS-1)+INODP)
   30   CONTINUE
        DO 40 INODL=1,NNODL
        SHAPL(INODL)=SHALG(NNODL*(IGAUS-1)+INODL)
        DO 40 IDIME=1,2
        CARTL(IDIME,INODL)=CARLG(IDIME,NNODL*(IGAUS-1)+INODL)
   40   CONTINUE
          IELEM=KELEM
          UVELY=0.0
          RADUS=0.0
          VVELY=0.0
C** EVAL. RAD.&PREV. VELS AT GUASS POINTS
          DUDX=0.0
          DUDY=0.0
          DVDX=0.0
          DVDY=0.0
C
          DO 50 INODP=1,NNODP
          KPOIN=IABS(LNODS(KELEM,INODP))
          ITOTU=NADFM(KPOIN)
          ITOTV=ITOTU+NODFM(KPOIN)-2
          SHAPE=SHAPP(INODP)
          UVELY=UVELY+VARB2(ITOTU)*SHAPE
          RADUS=RADUS+COORD(KPOIN,2)*SHAPE
        VVELY=VVELY+VARB2(ITOTV)*SHAPE
          CARXI=CARTP(1,INODP)
          CARYI=CARTP(2,INODP)
          UVEL=VARB2(ITOTU)
          VVEL=VARB2(ITOTV)
          DUDX=DUDX+CARXI*UVEL
          DUDY=DUDY+CARYI*UVEL
```

```
          DVDX=DVDX+CARXI*VVEL
          DVDY=DVDY+CARYI*VVEL
C
   50 CONTINUE
C
          EPS=DSQRT(2.0*(DUDX**2+0.5*((DUDY+DVDX)**2)+DVDY**2))
C
          IF(EPS.LT.0.0001) EPS=0.0001
          EPSN(KELEM,KIGS)=EPS
C
          VISC(KIGS)=(SIGSTAR+BSTAR*EPS**(1.0/EN))/(1.732*EPS)
          VISCP(KELEM,KIGS)=VISC(KIGS)
C
          IF(IAXSY.EQ.1)DAREA=DAREA*RADUS
C*** PUT HEAT GEN.TERM/OR BODY FORCES IN TO LOCAL RHS VECTOR
C
          DO 60 INODP=1,NNODP
          ERHST(INODP)=ERHST(INODP)+SHAPP(INODP)*EPS*PECLET*DAREA
        !*EPS*VISC(KIGS)
   60 CONTINUE
          DO 90 ICON1=1,4
          DO 90 ICON2=1,2
          IROWU=(ICON1-1)*7+4*ICON2-3
          IROWV=IROWU+3-ICON2
          IROWT=IROWU+4-ICON2
          INODP=2*(ICON1-1)+ICON2
          SHAPI=SHAPP(INODP)
          CARXI=CARTP(1,INODP)
          CARYI=CARTP(2,INODP)
          IF(ICON2.EQ.1) IROWP=IROWU+1
          DO 80 JCON1=1,4
          JCOLP=(JCON1-1)*7+2
          KGALI=(KIGS-1)*NNODL+JCON1
C
C*** PUT PR. TERM IN MOM. EQN.
          FLUMX(IROWU,JCOLP)=FLUMX(IROWU,JCOLP)-CARXI*SHALG(KGALI)
        !*DAREA
          FLUMX(IROWV,JCOLP)=FLUMX(IROWV,JCOLP)-CARYI*SHALG(KGALI)
        !*DAREA
          DO 80 JCON2=1,2
          JCOLU=(JCON1-1)*7+4*JCON2-3
          JCOLV=JCOLU+3-JCON2
          JCOLT=JCOLU+4-JCON2
          JNODP=2*(JCON1-1)+JCON2
          SHAPJ=SHAPP(JNODP)
          CARXJ=CARTP(1,JNODP)
          CARYJ=CARTP(2,JNODP)
C*** PUT DIFFU.& CONVE. TERMS IN MOM. EQN
          DIFFU=(CARXI*2.0*CARXJ+CARYI*CARYJ)*VISC(KIGS)*DAREA
          CONVC=((UVELY*CARXJ+VVELY*CARYJ)*(SHAPI*DAREA))/SIGMA
          FLUMX(IROWU,JCOLU)=FLUMX(IROWU,JCOLU)+DIFFU+CONVC
          DIFFU=(CARXI*CARXJ+2.0*CARYI*CARYJ)*VISC(KIGS)*DAREA
          FLUMX(IROWV,JCOLV)=FLUMX(IROWV,JCOLV)+DIFFU+CONVC
          FLUMX(IROWU,JCOLV)=CARYI*CARXJ*VISC(KIGS)*DAREA
        !+FLUMX(IROWU,JCOLV)
```

```
            FLUMX(IROWV,JCOLU)=FLUMX(IROWV,JCOLU)+CARXI*CARYJ*VISC(KIGS)
          !*DAREA
C*** FORM ENERGY EQN
C****PUT COND. &CONV. TERMS IN ENERGY EQN
            ENCOND=(CARXI*CARXJ+CARYI*CARYJ)*DAREA*THKSTAR
            ENCONV=(UVELY*CARXJ+VVELY*CARYJ)*SHAPI*DAREA*PECLET
          !*CPSTAR
            FLUMX(IROWT,JCOLT)=FLUMX(IROWT,JCOLT)+ENCOND+ENCONV
            IF(ICON2.EQ.2) GOTO 70
C*** FORM CONTINUITY EQN
            FLUMX(IROWP,JCOLU)=FLUMX(IROWP,JCOLU)+SHAPL(ICON1)*DAREA*CARXJ
            FLUMX(IROWP,JCOLV)=FLUMX(IROWP,JCOLV)+SHAPL(ICON1)*DAREA*CARYJ
            FLUMX(IROWP,JCOLP)=0.0
            FLUMX(IROWP,JCOLT)=0.0
   70 CONTINUE
   80 CONTINUE
   90 CONTINUE
            ENDDO
999    CONTINUE
C*************ADD LOCAL RHS VECTOR TO GLOBAL ARRAY
            DO 110 INODP=1,NNODP
            KINP=INODP
            KPOIN=IABS(LNODS(IELEM,KINP))
            ITOTU=NADFM(KPOIN)
            ITOTV=ITOTU+NODFM(KPOIN)-2
            ITOTT=ITOTU+NODFM(KPOIN)-1
            EQRHS(ITOTU)=EQRHS(ITOTU)+ERHSU(INODP)
            EQRHS(ITOTV)=EQRHS(ITOTV)+ERHSV(INODP)
            EQRHS(ITOTT)=EQRHS(ITOTT)+ERHST(INODP)
  110 CONTINUE
            DO 120 I=1,MBOUEL
            JK=I
            IF(KELEM.NE.IBEL(JK)) GOTO 120
            KBEL=IBEL(JK)
            KSURF=ISURF(JK)
C
            IF((KSURF.EQ.5).OR.(KSURF.EQ.6))THEN
            DO 1001 ICON1=1,4
            DO 1001 ICON2=1,2
            IROWV=IROWU+3-ICON2
            INODP=2*(ICON1-1)+ICON2
            IF(INODP.LE.3) THEN
            DO 1002 JCON1=1,4
            JCOLP=(JCON1-1)*7+2
            DO 1003 JCON2=1,2
            JCOLU=(JCON1-1)*7+4*JCON2-3
            JCOLV=JCOLU+3-JCON2
            JCOLT=JCOLU+4-JCON2
            JNODP=2*(JCON1-1)+JCON2
            FLUMX(IROWV,JCOLU)=SINA*FLUMX(IROWU,JCOLU)+
          !COSA*FLUMX(IROWV,JCOLU)
            FLUMX(IROWV,JCOLV)=SINA*FLUMX(IROWU,JCOLV)+
          !COSA*FLUMX(IROWV,JCOLV)
            FLUMX(IROWV,JCOLT)=SINA*FLUMX(IROWU,JCOLT)+
          !COSA*FLUMX(IROWV,JCOLT)
```

```
 1003 CONTINUE
          FLUMX(IROWV,JCOLP)=SINA*FLUMX(IROWU,JCOLP)+
     !COSA*FLUMX(IROWV,JCOLP)
 1002 CONTINUE
          ENDIF
 1001 CONTINUE
          ENDIF
          IF((KSURF.LE.3).OR.(KSURF.EQ.7))GOTO 120
          KSIDE=ISIDE(JK)
          CALL BOUCON(KBEL,KSURF,KSIDE,FLUMX,EQRHS,VARB2,MELEM,
     !MPOIN,NNODP,MTOTV,IITER)
  120 CONTINUE
          RETURN
          END
C*******************************************************************
C
C
          SUBROUTINE DRIVES(COORD,LNODS,MELEM,MPOIN,NELEM,NGAUS,
     !NNODL,NNODP,IELEM,AREAW)
          IMPLICIT REAL*8(A-H,O-Z)
C************************
C  EXT SUBROUTINES
C  DJACOB
          DIMENSION COORD(386,2),LNODS(111,8)
          DIMENSION DERIV(2,8),DJACI(2,2),DJACK(2,2),SHAPE(8)
          DIMENSION CARTP(2,8),AREAW(9)
          COMMON/ARC1/POSGP(3),WEIGP(3)
          COMMON/ AREA3/CARPG(2,72),SHAPG(72),CARLG(2,36),SHALG(36)
C         COMMON/AREA4/AREAW(9)
C  SET UP POSITIONS & WTS FOR  3 POINT GUASS RULE
          POSGP(1)=0.7745966692
          POSGP(2)=0.0
          POSGP(3)=-POSGP(1)
          WEIGP(1)=0.5555555556
          WEIGP(2)=0.8888888889
          WEIGP(3)=WEIGP(1)
C  CALCULATE SHAPE FUNCTIONS & DERIVATIVES FOR ELEMENTS
          LGAUS=0
          NGAUS=3
          DO 50 IGAUS=1,NGAUS
          DO 50 JGAUS=1,NGAUS
          LGAUS=LGAUS+1
          XEQIV=POSGP(IGAUS)
          YEQIV=POSGP(JGAUS)
C USE GAUS POSITIONS TO CALCULATE LOCAL VALUES
          CALL SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
C SET UP JACOBIAN MATRIX & INVERSE
          CALL DJACOB(DERIV,DETJB,DJACI,DJACK,IELEM,
     !MELEM,MPOIN,NNODP,LGAUS)
C CALCULATE GLOBAL DER & AREA*GAUS WTS
          DO 10 IDIME=1,2
          DO 10 INODP=1,NNODP
          CARTP(IDIME,INODP)=0.0
          DO 10 JDIME=1,2
          CARTP(IDIME,INODP)=CARTP(IDIME,INODP)+DJACI(IDIME,JDIME)
```

```
     !*DERIV(JDIME,INODP)
   10 CONTINUE
        AREAW(LGAUS)=DETJB*WEIGP(IGAUS)*WEIGP(JGAUS)
C PUT SHAPE FNS & DERIVS IN ELEMENT MX
        DO 20 INODP=1,NNODP
        KAPA=(LGAUS-1)*NNODP+INODP
        SHAPG(KAPA)=SHAPE(INODP)
        DO 20 IDIME=1,2
        CARPG(IDIME,KAPA)=CARTP(IDIME,INODP)
   20 CONTINUE
C USE GAUSS POSITIONS TO CALCULATE LOCAL VALUES
        CALL SHAPE4(DERIV,SHAPE,XEQIV,YEQIV)
C CALCULATE GLOBAL DERIVATIVES FOR LINEAR FUNCTIONS
        DO 30 IDIME=1,2
        DO 30 INODL=1,NNODL
        CARTP(IDIME,INODL)=0.0
        DO 30 JDIME=1,2
        CARTP(IDIME,INODL)=CARTP(IDIME,INODL)+
     !DJACI(IDIME,JDIME)*DERIV(JDIME,INODL)
   30 CONTINUE
C PUT SHAPE FNS & DERIVS IN ELEMENT MX
        DO 40 INODL=1,NNODL
        KGALI=(LGAUS-1)*NNODL+INODL
        SHALG(KGALI)=SHAPE(INODL)
   40 CONTINUE
   50 CONTINUE
        RETURN
        END
C*******************************************************************
C
        SUBROUTINE SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
        IMPLICIT REAL*8(A-H,O-Z)
        DIMENSION DERIV(2,8),SHAPE(8)
C PARABOLIC ELEMENT ANTICLOCKWISE NODE NUMBERING
        X=XEQIV
        Y=YEQIV
        XY=X*Y
        XX=X*X
        YY=Y*Y
        XXY=XX*Y
        XYY=X*YY
        X2=X*2.
        Y2=Y*2.
        XY2=XY*2.
      SHAPE(1)=(-1.+XY+XX+YY-XXY-XYY)*.25
        SHAPE(2)=(1.-Y-XX+XXY)*.5
        SHAPE(3)=(-1.-XY+XX+YY-XXY+XYY)*.25
        SHAPE(4)=(1.+X-YY-XYY)*.5
        SHAPE(5)=(-1.+XY+XX+YY+XXY+XYY)*.25
        SHAPE(6)=(1.+Y-XX-XXY)*.5
        SHAPE(7)=(-1.-XY+XX+YY+XXY-XYY)*.25
        SHAPE(8)=(1.-X-YY+XYY)*.5
        DERIV(1,1)=(Y+X2-XY2-YY)*.25
        DERIV(1,2)=-X+XY
        DERIV(1,3)=(-Y+X2-XY2+YY)*.25
```

```
      DERIV(1,4)=(1.-YY)*.5
      DERIV(1,5)=(Y+X2+XY2+YY)*.25
      DERIV(1,6)=-X-XY
      DERIV(1,7)=(-Y+X2+XY2-YY)*.25
      DERIV(1,8)=(-1.+YY)*.5
      DERIV(2,1)=(X+Y2-XX-XY2)*.25
      DERIV(2,2)=(-1.+XX)*.5
      DERIV(2,3)=(-X+Y2-XX+XY2)*.25
      DERIV(2,4)=-Y-XY
      DERIV(2,5)=(X+Y2+XX+XY2)*.25
      DERIV(2,6)=(1.-XX)*.5
      DERIV(2,7)=(-X+Y2+XX-XY2)*.25
      DERIV(2,8)=-Y+XY
      RETURN
      END
C*****************************************************************
C
      SUBROUTINE DJACOB(DERIV,DETJB,DJACI,DJACK,IELEM,
     !MELEM,MPOIN,NNODP,LGAUS)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION DERIV(2,8),DJACI(2,2),DJACK(2,2)
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
C SETUP TEMP MX TO ALLOW JACOBIIAN TO BE FORMED
      DO IDIME=1,2
      DO JDIME=1,2
      TEMPY=0.
      DO INODP=1,NNODP
      KPOIN=IABS(LNODS(IELEM,INODP))
C     KAG=(LGAUS-1)*NNODP+INODP
      XX=DERIV(IDIME,INODP)*COORD(KPOIN,JDIME)
      TEMPY=TEMPY+XX
      ENDDO
      DJACK(IDIME,JDIME)=TEMPY
      ENDDO
      ENDDO
      DETJB=DJACK(1,1)*DJACK(2,2)-DJACK(1,2)*DJACK(2,1)
C CHECK FOR NEG OR ZERO DET
      IF(DETJB)30,30,40
   30 CONTINUE
      TYPE*,DETJB,'DETJB'
      WRITE(*,2000)IELEM,LGAUS
      TYPE*,DJACK(1,1),DJACK(2,2),DJACK(1,2),DJACK(2,1)
      STOP
   40 CONTINUE
C INVERSE TEMP MX TO FORM JACOBIAN
      DJACI(1,1)=DJACK(2,2)/DETJB
      DJACI(2,2)=DJACK(1,1)/DETJB
      DJACI(1,2)=-DJACK(1,2)/DETJB
      DJACI(2,1)=-DJACK(2,1)/DETJB
 2000 FORMAT(//37H NON POSITIVE DETERMINENT FOR, ELEMENT,2I4)
      RETURN
      END
C*****************************************************************
C
      SUBROUTINE SHAPE4(DERIV,SHAPE,XEQIV,YEQIV)
```

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION DERIV(2,4),SHAPE(4)
C**LIN ELEMENT ANTICL NODE NUMBERING
      X=XEQIV
      Y=YEQIV
      XY=X*Y
      SHAPE(1)=(1.-X-Y+XY)*.25
      SHAPE(2)=(1.+X-Y-XY)*.25
      SHAPE(3)=(1.+X+Y+XY)*.25
      SHAPE(4)=(1.-X+Y-XY)*.25
      DERIV(1,1)=(-1.+Y)*.25
      DERIV(1,2)=(1.-Y)*.25
      DERIV(1,3)=(1.+Y)*.25
      DERIV(1,4)=(-1.-Y)*.25
      DERIV(2,1)=(-1.+X)*.25
      DERIV(2,2)=(-1.-X)*.25
      DERIV(2,3)=(1.+X)*.25
      DERIV(2,4)=(1.-X)*.25
      RETURN
      END
C*************************************************************
C
      SUBROUTINE BOUCON(KBEL,KSURF,KSIDE,FLUMX,EQRHS,VARB2,
     !MELEM,MPOIN,NNODP,MTOTV,IITER)
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/ARC1/POSGP(3),WEIGP(3)
      COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
      COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
      COMMON/ARC13/RHO,MBOUEL
      COMMON/ARC30/ANGLN(386)
     COMMON/ARC14/IBEL(60),ISURF(60),ISIDE(60)
      DIMENSION EQRHS(2000),VARB2(2000),FLUMX(28,28)
      DIMENSION M(3),NODE(3),DERIV(2,8),SHAP1(8),SHAPE(8)
      DIMENSION RHSU(8),RHSV(8),RHST(8),CART(2,8),DERX(2,8)
      DIMENSION CJACK(2,2),CJACI(2,2),COSN(3),SINN(3)
      COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
     !BSTAR,SIGYO,SPHTO,THKO,SIGMA
      COMMON/ARC887/SLETHG(111,3)
C** IDENTIFYING LOCAL NODE NOS
      IF(KSURF.LE.3)GOTO 88
      M(1)=1
      M(2)=5
      M(3)=8
      NODE(1)=2*KSIDE-1
      NODE(2)=NODE(1)+1
      NODE(3)=NODE(2)+1
      IF(NODE(3).GT.8) NODE(3)=NODE(3)-8
      DO 145 I=1,8
      RHSU(I)=0.0
      RHSV(I)=0.0
      RHST(I)=0.0
  145 CONTINUE
C-------------------------------------
C GAUSS POINT EVAL STARTS
C-------------------------------------
```

```
      LGAUS=0
      DO 150 IGAUS=1,3
      XEQIV=POSGP(IGAUS)
      YEQIV=-1.0
C
      CALL SHAPE8(DERX,SHAP1,XEQIV,YEQIV)
      TEMPR=0.
      DO INODP=1,8
      KPOIN=IABS(LNODS(KBEL,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTT=ITOTU+NODFM(KPOIN)-1
      TEMPR=TEMPR+VARB2(ITOTT)*SHAP1(INODP)
      ENDDO
C
      CALL PROP(TEMPR,SIGSTAR,CPSTAR,THKSTAR,SIGMAY)
C
      DO KINP=1,8
      KPOIN=IABS(LNODS(KBEL,KINP))
      ITOTU=NADFM(KPOIN)
      ITOTT=ITOTU+NODFM(KPOIN)-1
      LGAUS=LGAUS+1
      GOTO (10,20,30,40) , KSIDE
C** CAL LENGTH OF ELEM AT EACH GAUSS PT
   10 CONTINUE
      XEQIV=POSGP(IGAUS)
      YEQIV=-1.0
      ITEMP=1
      GOTO 50
   20 CONTINUE
      XEQIV=1.0
      YEQIV=POSGP(IGAUS)
      ITEMP=2
      GOTO 50
   30 CONTINUE
      XEQIV=POSGP(IGAUS)
      YEQIV=1.0
      ITEMP=1
      GOTO 50
   40 CONTINUE
      XEQIV=-1.0
      YEQIV=POSGP(IGAUS)
      ITEMP=2
   50 CONTINUE
      CALL SHAPE3(DERX,SHAP1,XEQIV,YEQIV)
      CALL SHAPE4(DERIV,SHAPE,XEQIV,YEQIV)
      IELEM=KBEL
      MPOIN=386
      MELEM=111
      NNODP=8
      CALL DJACOB(DERX,DETJB,CJACI,CJACK,IELEM,
     !MELEM,MPOIN,NNODP,LGAUS)
      TEMPY=DSQRT(CJACK(ITEMP,1)**2+CJACK(ITEMP,2)**2)
      SLETH=TEMPY*WEIGP(IGAUS)
      SLETHG(KBEL,IGAUS)=SLETH
C** CAL VISC & STRAIN RATE AT GAUSS PT FOR S TH SURFACE ONLY
```

```
      IF(KSURF.NE.5) GOTO 15
      DO 51 IDIME=1,2
      DO 51 INODP=1,NNODP
      CART(IDIME,INODP)=0.0
      DO 51 JDIME=1,2
      CART(IDIME,INODP)=CART(IDIME,INODP)+DERX(JDIME,INODP)
     !*CJACI(IDIME,JDIME)
   51 CONTINUE
      UVELY=0.0
      VVELY=0.0
      DUDX=0.0
      DUDY=0.0
      DVDX=0.0
      DVDY=0.0
      DO 60 I=1,3
      KPOIN=IABS(LNODS(KBEL,I))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-2
      SHAP=SHAP1(I)
      UVELY=UVELY+VARB2(ITOTU)*SHAP
      VVELY=VVELY+VARB2(ITOTV)*SHAP
      CARXI=CART(1,I)
      CARYI=CART(2,I)
      UVEL=VARB2(ITOTU)
      VVEL=VARB2(ITOTV)
      DUDX=DUDX+CARXI*UVEL
      DUDY=DUDY+CARYI*UVEL
      DVDX=DVDX+CARXI*VVEL
      DVDY=DVDY+CARYI*VVEL
   60 CONTINUE
      EPS=DSQRT(2.0*(DUDX**2+0.5*(DUDY+DVDX)**2+DVDY**2))
      IF(EPS.LT.0.001) EPS=0.001
      VISC=(SIGSTAR+BSTAR*EPS**(1.0/EN))/(1.732*EPS)
      VT=VVELY*COSA+UVELY*SINA
C
C EVAL GAUSS PT INTEGRALS FOR LOCAL RHS VECTOR
      DO 13 I=1,3
      NOD=NODE(I)
      SHAP=SHAP1(NOD)
      FACTOR=FRIFAC*VISC*EPS
      RHSV(NOD)=RHSV(NOD)-SHAP*FACTOR*SLETH
      RHST(NOD)=RHST(NOD)+SHAP*PECLET*FACTOR*VT*CONRAT*SLETH/THKSTAR
      RHSU(NOD)=0.
   13 CONTINUE
C** SKIPPING CONV BOUN CON FOR 5 TH SUR
      IF(KSURF.EQ.5)GOTO 150
   15 CONTINUE
C IMPOSING SHEAR STRESS BC FOR 6 TH SUR
      IF(KSURF.EQ.6)THEN
      IPOINA=116
      IPOINB=122
      SLOPE=FACTOR/(COORD(IPOINB,1)-COORD(IPOINA,1))
C
      DO IJ=1,3
      NODD=NODE(IJ)
```

```
          SHAP=SHAP1(NODD)
          IPOIN=IABS(LNODS(KBEL,NODD))
          XDIST=COORD(IPOINB,1)-COORD(IPOIN,1)
          FACTOR=SLOPE*XDIST
          RHSV(NODD)=RHSV(NODD)-SHAP*FACTOR*SLETH
          RHST(NODD)=RHST(NODD)+SHAP*PECLET*FACTOR*VT*CONRAT*SLETH/THKSTAR
          RHSU(NODD)=0.
          ENDDO
          ENDIF
C
C** INCOR APPR CONV HT TR BC
          DO 75 II=1,3
          INODP=2*(KSIDE-1)+II
          IROWU=7*(KSIDE-1)+M(II)
          IF(INODP.GT.8) INODP=INODP-8
          IF(IROWU.GT.28) IROWU=IROWU-28
          KPOIN=IABS(LNODS(KBEL,INODP))
          IROWT=IROWU+NODFM(KPOIN)-1
          SHAPI=SHAP1(INODP)
          RHST(INODP)=RHST(INODP)+SHAPI*HSTAR*TFSTAR*SLETH/THKSTAR
          DO 95 JCON1=1,4
          JCOLP=(JCON1-1)*7+2
          DO 95 JCON2=1,2
          JNODP=(JCON1-1)*2+JCON2
          JCOLU=(JCON1-1)*7+JCON2*4-3
          JCOLT=(JCON1-1)*7+JCON2*3+1
          SHAPJ=SHAP1(JNODP)
          TERM=HSTAR*SHAPI*SHAPJ*SLETH/THKSTAR
          FLUMX(IROWT,JCOLT)=FLUMX(IROWT,JCOLT)+TERM
   95 CONTINUE
   75 CONTINUE
          ENDDO
  150 CONTINUE
   26 CONTINUE
C ** INCOR NORMAL VEL BDC
          ANGLN(100)=90.0
          DO 80 II=1,3
C** EVAL DIR OF NORMAL AT EACH NODE
          GOTO(100,110,120,130),KSIDE
  100 CONTINUE
          XEQIV=FLOAT(II)-2.0
          YEQIV=-1.0
          ITEMP=1
          RTEMP=+1.0
          GOTO 140
  110 CONTINUE
          XEQIV=1.0
          YEQIV=FLOAT(II)-2.0
          ITEMP=2
          RTEMP=1.0
          GOTO 140
  120 CONTINUE
          XEQIV=-FLOAT(II)+2.0
          YEQIV=1.0
          ITEMP=1
```

```
        RTEMP=-1.0
        GOTO 140
  130 CONTINUE
        XEQIV=-1.0
        YEQIV=-FLOAT(II)+2.0
        ITEMP=2
        RTEMP=-1.0
  140 CONTINUE
        CALL SHAPE8(DERX,SHAP1,XEQIV,YEQIV)
C** EVAL SLOPE AT EACH NODE
        INODP=2*(KSIDE-1)+II
        IROWU=7*(KSIDE-1)+M(II)
        IF(INODP.GT.8) INODP=INODP-8
        IF(IROWU.GT.28) IROWU=IROWU-28
        KPOIN=IABS(LNODS(KBEL,INODP))
        IROWV=IROWU+NODFM(KPOIN)-2
        IROWT=IROWV+1
        IF(KSURF.EQ.5) ANGLN(100)=RAKE
        ANGLE=ANGLN(KPOIN)
        ANGLE=ANGLE*PI/180.0
        COSLX=RTEMP*DCOS(ANGLE)
        COSN(II)=COSLX
        IF(DABS(COSN(II)).LT.1.0E-4) COSN(II)=0.0
        COSLY=-RTEMP*DSIN(ANGLE)
        SINN(II)=COSLY
        IF(DABS(SINN(II)).LT.1.0E-4) SINN(II)=0.0
        DO 90 JCON1=1,4
        JCOLP=(JCON1-1)*7+2
        DO 90 JCON2=1,2
        JNODP=(JCON1-1)*2+JCON2
        JCOLU=(JCON1-1)*7+JCON2*4-3
        JCOLV=JCOLU+3-JCON2
        JCOLT=JCOLU+4-JCON2
        SHAPPJ=SHAP1(JNODP)
C**DET APPR MOM EQN FOR NORMAL VEL BC
        IF(DABS(COSN(II)).GE.DABS(SINN(II))) GOTO 55
        FLUMX(IROWV,JCOLU)=COSN(II)*SHAPPJ
        FLUMX(IROWV,JCOLV)=SINN(II)*SHAPPJ
        FLUMX(IROWV,JCOLT)=0.0
        IF(JCON2.EQ.2) GOTO 56
        FLUMX(IROWV,JCOLP)=0.0
   56   RHSV(INODP)=0.0
        GOTO 90
   55 CONTINUE
        FLUMX(IROWU,JCOLU)=COSN(II)*SHAPPJ
        FLUMX(IROWU,JCOLV)=SINN(II)*SHAPPJ
        FLUMX(IROWU,JCOLT)=0.0
        IF(JCON2.EQ.2) GOTO 67
        FLUMX(IROWU,JCOLP)=0.0
   67 RHSU(INODP)=0.0
   90 CONTINUE
   80 CONTINUE
C** ASS GAUSS PT INTEGRALS IN GLOBAL RHS VECTOR
        DO 200 II=1,3
        NOD=NODE(II)
```

```
      KPOIN=IABS(LNODS(KBEL,NOD))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-2
      ITOTT=ITOTU+NODFM(KPOIN)-1
      EQRHS(ITOTU)=EQRHS(ITOTU)+RHSU(NOD)
      EQRHS(ITOTV)=EQRHS(ITOTV)+RHSV(NOD)
      EQRHS(ITOTT)=EQRHS(ITOTT)+RHST(NOD)
  200 CONTINUE
   88 RETURN
      END
C
C*************************************************************************
C
      SUBROUTINE FORCE
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION DERIV(2,8),DJACI(2,2),DJACK(2,2),SHAPE(8)
      DIMENSION CARTP(2,8),AREAW(9),CART(2,10)
      DIMENSION CJACK(2,2),CJACI(2,2),COSN(3),SINN(3)
      DIMENSION M(3),NODE(3),SHAP1(8)
     COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
     COMMON/ARC13/RHO,MBOUEL
       COMMON/ARC881/FW(111,3)
     COMMON/ARC10/CUTVEL,DCUT,WID,RAKE
     COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
     COMMON/ARC3/VARB1(2000),VARB2(2000)
       COMMON/ARC1/POSGP(3),WEIGP(3)
       COMMON/ AREA3/CARPG(2,72),SHAPG(72),CARLG(2,36),SHALG(36)
       COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
    !BSTAR,SIGYO,SPHTO,THKO,SIGMA
C
C  SET UP POSITIONS & WTS FOR  3 POINT GUASS RULE
C
      POSGP(1)=0.7745966692
      POSGP(2)=0.0
      POSGP(3)=-POSGP(1)
      WEIGP(1)=0.5555555556
      WEIGP(2)=0.8888888889
      WEIGP(3)=WEIGP(1)
      KELEM=33
      TFORCEL=0.
      FFORCEL=0.
      CONLEN=0.
      XFOR=0.
      YFOR=0.
      XFOR1=0.
      YFOR1=0.
      DO 77 IE=1,8
      KELEM=KELEM+1
      LGAUS=0
      NGAUS=3
      TFORCE=0.
      FFORCE=0.
      F=0.
      FF=0.
      FNN=0.
```

```
C CALCULATIONS START FOR EACH G.PT.
      DO 50 IGAUS=1,NGAUS
C SETTING UP GAUS POINT POSITIONS.
      LGAUS=LGAUS+1
      XEQIV=POSGP(IGAUS)
      YEQIV=-1.0
C
C CALCULATING PRESSURE FOR G.PT.
      CALL SHAPE4(DERIV,SHAPE,XEQIV,YEQIV)
      TP=0.
      DO 40 INODL=1,4
      INODP=(INODL-1)*2+1
      KPOIN=IABS(LNODS(KELEM,INODP))
      IODFM=NODFM(KPOIN)
      IADFM=NADFM(KPOIN)
     P=VARB1(IADFM+1)
      TP=TP+P*SHAPE(INODL)
   40 CONTINUE
C
      CALL SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
      IELEM=KELEM
      MELEM=111
      MPOIN=386
      NNODP=8
      CALL DJACOB(DERIV,DETJB,CJACI,CJACK,IELEM,
     !MELEM,MPOIN,NNODP,LGAUS)
C FINDING TEMPR. & CALLING PROP VAR.
     TEMPR=0.
      XGAUS=0.
      YGAUS=0.
      DO 60 INODP=1,8
      KPOIN=IABS(LNODS(KELEM,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTT=ITOTU+NODFM(KPOIN)-1
      TEMPR=TEMPR+VARB2(ITOTT)*SHAPE(INODP)
C FINDIND GLOBAL POS. OF G.PTS
      XGAUS=XGAUS+(SHAPE(INODP)*COORD(KPOIN,1))*DCUT*1000.
      YGAUS=YGAUS+(SHAPE(INODP)*COORD(KPOIN,2))*DCUT*1000.
   60 CONTINUE
C  CALLING PROP. VAR FOR TOTL G PT. TEMPR
      CALL PROP(TEMPR,SIGSTAR,CPSTAR,THKSTAR,SIGMAY)
C
      DO 51 IDIME=1,2
      DO 51 INODP=1,8
      CART(IDIME,INODP)=0.0
      DO 51 JDIME=1,2
      CART(IDIME,INODP)=CART(IDIME,INODP)+DERIV(JDIME,INODP)
     !*CJACI(IDIME,JDIME)
   51 CONTINUE
C
C FINDING VISC.,DUDX ETC.,
C
      DUDX=0.0
      DUDY=0.0
      DVDX=0.0
```

```
      DVDY=0.0
      UVELY=0.
      VVELY=0.
      DO 91 INODP=1,8
      KPOIN=IABS(LNODS(KELEM,INODP))
      ITOTU=NADFM(KPOIN)
      ITOTV=ITOTU+NODFM(KPOIN)-2
      CARXI=CART(1,INODP)
      CARYI=CART(2,INODP)
      UVEL=VARB2(ITOTU)
      VVEL=VARB2(ITOTV)
      UVELY=UVELY+VARB2(ITOTU)*SHAPE(INODP)
      VVELY=VVELY+VARB2(ITOTV)*SHAPE(INODP)
      DUDX=DUDX+CARXI*UVEL
      DUDY=DUDY+CARYI*UVEL
      DVDX=DVDX+CARXI*VVEL
      DVDY=DVDY+CARYI*VVEL
   91 CONTINUE
C
      EPS=DSQRT(2.0*(DUDX**2+0.5*(DUDY+DVDX)**2+DVDY**2))
      IF(EPS.LT.0.0001) EPS=0.0001
      VISC=(SIGSTAR+BSTAR*EPS**(1.0/EN))/(1.732*EPS)
      VT=VVELY*COSA+UVELY*SINA
      FW(KELEM,IGAUS)=FRIFAC*VISC*EPS*VT
C
C NOW CAL. STRESSES & FORCES USING G.PT VALUES
      SIGXX=-TP+2.*VISC*DUDX
      SIGYY=-TP+2.*VISC*DVDY
      SIGXY=VISC*(DUDY+DVDX)
      FX=(-SIGXX*COSA+SIGXY*SINA)
      FY=(-SIGXY*COSA+SIGYY*SINA)
      IPOIN1=IABS(LNODS(KELEM,1))
      IPOIN2=IABS(LNODS(KELEM,3))
      DEL=DSQRT((COORD(IPOIN2,1)-COORD(IPOIN1,1))**2+
     !(COORD(IPOIN2,2)-COORD(IPOIN1,2))**2)
      DEL=DEL/2.
      TFORCE=TFORCE+(FX*DEL*WEIGP(IGAUS)*WID/DCUT)
      FFORCE=FFORCE+(FY*DEL*WEIGP(IGAUS)*WID/DCUT)
C
C CAL. FRICTION & NORMAL FORCES
      F=F+FRIFAC*VISC*EPS*DEL*WEIGP(IGAUS)*WID/DCUT
      FACTOR=FRIFAC*VISC*EPS
C CAL. NORMAL  FORCE FROM DERIVED REL
      STNORM=(SIGXX*COSA**2+SIGYY*SINA**2+SIGXY*SIN2A)
      FNN=FNN+STNORM*DEL*WEIGP(IGAUS)*WID/DCUT
      STSHEA=((SIGYY-SIGXX)/2.*SIN2A+SIGXY*(1.-2.*SINA**2))
      FF=FF+STSHEA*DEL*WEIGP(IGAUS)*WID/DCUT
      STNORMD=STNORM*SIGYO
      STSHEAD=STSHEA*SIGYO
      WRITE(38,*)'KELEM    IGAUS   X-C     Y-C   NORM.STRESS   SHEAR STRESS'
      WRITE(38,*)'######    ######   ###   ########   ############## ############'
      WRITE(38,388)KELEM,IGAUS,XGAUS,YGAUS,STNORMD,FACTOR
  388 FORMAT(/I3,4X,I2,2X,D8.2,X,D8.2,D12.4,2X,D12.4)
   50 CONTINUE
      XFOR=XFOR+(-F*SINA-FNN*COSA)
```

```
      YFOR=YFOR+(-F*COSA+FNN*SINA)
      XFOR1=XFOR1+(-FF*SINA-FNN*COSA)
      YFOR1=YFOR1+(-FF*COSA+FNN*SINA)
C
      TFORCEL=TFORCEL+TFORCE
      FFORCEL=FFORCEL+FFORCE
   77 CONTINUE
      XFOR=XFOR*SIGYO*DCUT*DCUT
      YFOR=YFOR*SIGYO*DCUT*DCUT
      XFOR1=XFOR1*SIGYO*DCUT*DCUT
      YFOR1=YFOR1*SIGYO*DCUT*DCUT
C
      TFORCEL=TFORCEL*SIGYO*DCUT*DCUT
      FFORCEL=FFORCEL*SIGYO*DCUT*DCUT
C
      WRITE(35,199) TFORCEL,FFORCEL,XFOR,YFOR,XFOR1,YFOR1
  199 FORMAT(//8X,'TANGENTIAL  FORCE IS        ',D12.4,' N',
     !        /8X,'FEED FORCE  IS              ',D12.4,' N',
     !        /8X,'TANG. FORCE  IS(FROM F&N)   ',D12.4,' N',
     !        /8X,'FEED FORCE  IS(FROM F&N)    ',D12.4,' N',
     !        /8X,'TANG. FORCE  IS(FROM FF&N)  ',D12.4,' N',
     !        /8X,'FEED FORCE  IS(FROM FF&N)   ',D12.4,' N')
      RETURN
      END
C ***************************************************************
C
      SUBROUTINE PROP(TEMPR,SIGSTAR,CPSTAR,THKSTAR,SIGMAY)
      IMPLICIT REAL*8(A-H,O-Z)
     COMMON/ARC13/RHO,MBOUEL,THKTOL
     COMMON/ARC10/CUTVEL,DCUT,WID,RAKE,GAMMA
     COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
      COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
     !BSTAR,SIGYO,SPHTO,THKO,SIGMA
      TEMPR=TEMPR*TREF
      IF(TEMPR.GT.373.)GOTO 88
      SPHT=SPHTO
      SIGMAY=SIGYO
      THK=THKO
      GOTO 10
   88 IF((TEMPR.GT.373.).AND.(TEMPR.LE.1173.))THEN
      AM=-0.02226
      AC=50.
      ELSE
      IF(TEMPR.GT.1173)THEN
      AM=0.00466666
      AC=29.63
      ENDIF
      ENDIF
      THK=AM*TEMPR+AC
      IF((TEMPR.GT.373.).AND.(TEMPR.LE.473.))THEN
      OMX=0.03185
      OCX=488.119
      ELSE
      IF((TEMPR.GT.473.).AND.(TEMPR.LE.773.))THEN
      OMX=0.5859
```

```
          OCX=229.239
          ELSE
          IF((TEMPR.GT.773.).AND.(TEMPR.LE.973.))THEN
          OMX=1.00435
          OCX=-94.222
          ELSE
          IF((TEMPR.GT.973.).AND.(TEMPR.LE.1173.))THEN
          OMX=-0.64865
          OCX=1514.146
          ELSE
          IF((TEMPR.GT.1173.).AND.(TEMPR.LE.1373.))THEN
          OMX=0.0209
          OCX=728.76
          ELSE
          IF((TEMPR.GT.1373.).AND.(TEMPR.LE.1473.))THEN
          OMX=0.4604
          OCX=125.33
          ENDIF
          ENDIF
          ENDIF
          ENDIF
          ENDIF
          ENDIF
        SPHT=OMX*TEMPR+OCX
C         A=2.5133E08
C         B=191784.9
C         CS=-366.8211
          A=9.6991731E08
          B=-2137677.
          CS=1521.180
          SIGMAY=(A+B*TEMPR+CS*TEMPR**2)
   10   SIGSTAR=SIGMAY/SIGY0
          CPSTAR=SPHT/SPHT0
          THKSTAR=THK/THK0
          RETURN
          END
C*****************************************************************
C*****************************************************************
          SUBROUTINE STRATES
          IMPLICIT REAL*8(A-H,O-Z)
          DIMENSION DUDX(386),DUDY(386),DVDX(386),DVDY(386)
          DIMENSION NCOUNT(1000),XP(8),YP(8)
          COMMON/ARC992/STRATE(386)
          DIMENSION DERIV(2,8),DJACI(2,2),DJACK(2,2),SHAPE(8)
          DIMENSION CARTP(2,8),AREAW(9),CART(2,10)
          DIMENSION CJACK(2,2),CJACI(2,2),COSN(3),SINN(3)
        COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
        COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
        COMMON/ARC3/VARB1(2000),VARB2(2000)
          COMMON/ARC1/POSGP(3),WEIGP(3)
          DATA XP/-1.0,0.,1.0,1.0,1.0,0.,-1.,-1./
          DATA YP/-1.0,-1.0,-1.0,0.,1.,1.,1.,0./
          NNODP=8
          NELEM=111
C INITIALISING DUDX ETC.
```

```
      DO   INODP=1,NNODP
      DO   IEL=1,NELEM
      INOD=IABS(LNODS(IEL,INODP))
      DUDX(INOD)=0.0
      DUDY(INOD)=0.0
      DVDX(INOD)=0.0
      DVDY(INOD)=0.0
      NCOUNT(INOD)=0
      ENDDO
      ENDDO
C
      DO 200 IEL=1,NELEM
      DO 100 INODP=1,NNODP
      INOD=IABS(LNODS(IEL,INODP))
      ITOTU=NADFM(INOD)
      ITOTV=ITOTU+NODFM(INOD)-2
      XEQIV=XP(INODP)
      YEQIV=YP(INODP)
      CALL SHAPE8(DERIV,SHAPE,XEQIV,YEQIV)
      IELEM=IEL
      MELEM=111
      MPOIN=386
      CALL DJACOB(DERIV,DETJB,CJACI,CJACK,IELEM,
     !MELEM,MPOIN,NNODP)
C
      DO 51 IDIME=1,2
      DO 51 INDP=1,8
      CART(IDIME,INDP)=0.0
      DO 51 JDIME=1,2
      CART(IDIME,INDP)=CART(IDIME,INDP)+DERIV(JDIME,INDP)
     !*CJACI(IDIME,JDIME)
   51 CONTINUE
      CARXI=CART(1,INODP)
      CARYI=CART(2,INODP)
      UVEL=VARB2(ITOTU)
      VVEL=VARB2(ITOTV)
      DUDX(INOD)=DUDX(INOD)+CARXI*UVEL
      DUDY(INOD)=DUDY(INOD)+CARYI*UVEL
      DVDX(INOD)=DVDX(INOD)+CARXI*VVEL
      DVDY(INOD)=DVDY(INOD)+CARYI*VVEL
      NCOUNT(INOD)=NCOUNT(INOD)+1
  300 CONTINUE
  100 CONTINUE
  200 CONTINUE
C
      DO IPOIN=1,386
      DUDX(IPOIN)=DUDX(IPOIN)/NCOUNT(IPOIN)
      DUDY(IPOIN)=DUDY(IPOIN)/NCOUNT(IPOIN)
      DVDX(IPOIN)=DVDX(IPOIN)/NCOUNT(IPOIN)
      DVDY(IPOIN)=DVDY(IPOIN)/NCOUNT(IPOIN)
C
      STRATE(IPOIN)=DSQRT(2.0*(DUDX(IPOIN)**2
     !+0.5*(DUDY(IPOIN)+DVDX(IPOIN))**2+DVDY(IPOIN)**2))
      IF(STRATE(IPOIN).LT.0.0001) STRATE(IPOIN)=0.0001
      WRITE(90,*)IPOIN,STRATE(IPOIN)
```

```
      ENDDO
      RETURN
      END
C*****************************************
      SUBROUTINE CHTHK
      IMPLICIT REAL*8(A-H,O-Z)
      COMMON/ARC881/FW(111,3)
     COMMON/ARC12/FRIFAC,PI,EN,COSA,SINA,SIN2A
     COMMON/ARC13/RHO,MBOUEL
     COMMON/ARC14/IBEL(60),ISURF(60),ISIDE(60)
     COMMON/ARC10/CUTVEL,DCUT,RAKE
     COMMON/ARC2/LNODS(111,8),COORD(386,2),NADFM(386),NODFM(386)
     COMMON/ARC35/EPSN(111,9),XG(111,9),YG(111,9),VISCP(111,9)
    !,DAREAG(111,9)
      COMMON/ARC887/SLETHG(111,3)
      COMMON/ARC991/TREF,PECLET,CONRAT,TFSTAR,HSTAR,
    !BSTAR,SIGYO,SPHTO,THKO,SIGMA
C
      TPLASTWK=0.
      DO KELEM=1,111
C
      PLASTWKG=0.
      DO KIGS=1,9
C
      SIGFLO=1.732*VISCP(KELEM,KIGS)*EPSN(KELEM,KIGS)
      PLASTWK=SIGFLO*EPSN(KELEM,KIGS)
      PLASTWKG=TPLASTWK+PLASTWK*DAREAG(KELEM,KIGS)
      ENDDO
      TPLASTWK=TPLASTWK+PLASTWKG
      ENDDO
      TPLASTWKD=TPLASTWK
C
      TFWK=0.

      I=33
      DO IJ=1,8
      I=I+1
      FWKG=0.
C
      DO IGAUS=1,3
      FWK=FW(I,IGAUS)*SLETHG(I,IGAUS)
      FWKG=FWKG+FWK
      ENDDO
      TFWK=TFWK+FWKG
      ENDDO
      TFWKD=TFWK
C
      TWKD=TPLASTWKD+TFWKD
      CHIPTK=(COORD(122,1)-COORD(386,1))**2+
    !(COORD(386,2)-COORD(122,2))**2
      CHIPTK=DSQRT(CHIPTK)*DCUT*1000.
C
      WRITE(35,223)CHIPTK,TPLASTWKD,TFWKD,TWKD
223   FORMAT(///' CHIP THICKNESS      = ',D12.4,' MM ',
    !       /' TOTAL PLASTIC WORK = ',D12.4,'     ',
```

```
!        /' TOTAL FRICTION WORK =',D12.4,' MM ',
!        /' TOTAL WORK          = ',D12.4,' MM ')
      RETURN
      END
C ######################################################################
C ######################END#############################################
```

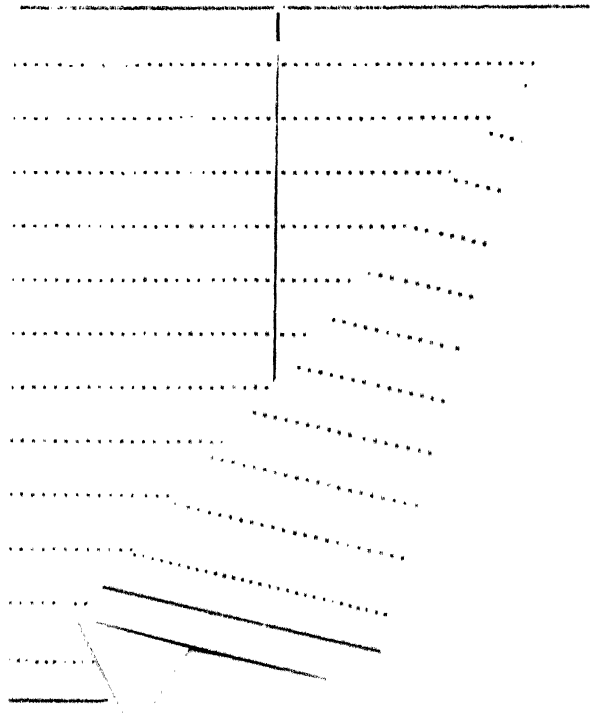
113088

TH
671.3
D459S

113088

Date Slip

This book is to be returned on the date last stamped.



ME-1992-M-DES-STU